

AIX Version 7.2

*Verwaltung von Betriebssystem und
Einheiten*

IBM

AIX Version 7.2

*Verwaltung von Betriebssystem und
Einheiten*

IBM

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen im Abschnitt „Bemerkungen“ auf Seite 695 gelesen werden.

Diese Ausgabe bezieht sich auf AIX Version 7.2 und alle nachfolgenden Releases und Modifikationen, bis dieser Hinweis in einer Neuauflage geändert wird.

© Copyright IBM Corporation 2015, 2017.

Inhaltsverzeichnis

Zu diesem Handbuch	v
Hervorhebung	v
Groß-/Kleinschreibung in AIX	v
ISO 9000.	v

Verwaltung von Betriebssystem und Einheiten. **1**

Neuerungen im Betriebssystem und bei der Einheitenverwaltung	1
Betriebssystemverwaltung	1
Verfügbare Schnittstellen für die Systemverwaltung	2
Elementare Softwaredaten	2
Betriebssystemaktualisierungen	3
Systemstart	4
Systemsicherung.	22
Systemabschluss durchführen	54
Systemumgebung	55
AIX Runtime Expert	68
Befehle und Prozesse	132
Systemblockierungen verwalten	154
Prozessverwaltung	157
Systemabrechnung	164
System Resource Controller.	192
Betriebssystemdateien	197
Betriebssystem-Shells	217
Sicherheit des Betriebssystems.	312
Benutzerumgebung	326
Referenz für BSD-Systeme	341
Eingabe- und Ausgabeumleitung	362
AIX-Kernelwiederherstellung	369
Einheitenmanagement	371
Logical Volume Manager	371
Speicherung logischer Datenträger	409
Paging-Bereich und virtueller Speicher	442
Dateisysteme	453

Workload Manager	514
Einheitenknoten	563
Positionscodes für Einheiten	565
iSCSI konfigurieren	568
Hot-Plug-Verwaltung von PCI-Einheiten	570
Multiple Path I/O	575
Zieleinheitenkonfiguration	599
Bandlaufwerke	600
Unterstützung von USB-Einheiten	613
Caching von Speicherdaten.	615
Anmeldenamen, System-IDs und Kennwörter	624
Common Desktop Environment	631
Drucken und Druckjobs	637
Live Partition Mobility with Host Ethernet Adapters	650
Adapter für DLPAR verlagern.	652
Loopback-Einheit	653
AIX Event Infrastructure for AIX and AIX Clusters - AHAFS	653
Einführung in AIX Event Infrastructure	653
Komponenten von AIX Event Infrastructure	654
AIX Event Infrastructure konfigurieren	656
Abstrakte Sicht der Funktionsweise von AIX Event Infrastructure	656
AIX Event Infrastructure verwenden	658
Ereignisse überwachen	658
Vordefinierte Ereigniserzeuger.	670
Clusterereignisse	683
Vordefinierte Ereigniserzeuger für eine cluster-sensitive AIX-Instanz	684
Bemerkungen	695
Hinweise zur Datenschutzrichtlinie	697
Marken	697
Index	699

Zu diesem Handbuch

Dieses Dokument enthält vollständige Informationen für Benutzer und Systemadministratoren, die die Auswahl von Optionen bei der Ausführung von Tasks beeinflussen können. Beispiele für solche Tasks sind das Sichern und Wiederherstellen des Systems, das Verwalten physischer und logischer Speicher, die Auswahl der passenden Größe für den Paging-Bereich usw. Außerdem enthält das Handbuch vollständige Informationen zum Ausführen von Tasks wie der Verwaltung von logischen Datenträgern, Speicher und Ressourcen. Systembenutzer können sich mit der Ausführung von Befehlen, dem Umgang mit Prozessen, der Bearbeitung von Dateien und Verzeichnissen und grundlegenden Druckaufgaben vertraut machen.

Zu den weiteren hilfreichen Themen für Benutzer und Systemadministratoren, die in diesem Handbuch behandelt werden, gehören das Erstellen und das Festlegen der Größe eines Paging-Bereichs, das Verwalten des virtuellen Speichers, das Sichern und Wiederherstellen des Systems, das Verwalten von Hardware- und Pseudoeinheiten, das Verwenden des System Resource Controller (SRC), das Schützen von Dateien, das Verwenden von Speichermedien, das Anpassen von Umgebungsdateien und das Schreiben von Shell-Scripts. Sie finden dieses Dokument auch auf der zum Betriebssystem gelieferten Dokumentations-CD.

Hervorhebung

In diesem Dokument werden die folgenden Hervorhebungskonventionen verwendet:

Fettschrift	Befehle, Subroutinen, Schlüsselwörter, Dateien, Strukturen, Verzeichnisse und andere Elemente, deren Namen vom System vorgegeben sind, werden in Fettschrift hervorgehoben. In Fettschrift werden auch grafische Objekte wie Schaltflächen, Beschriftungen und Symbole angegeben, die der Benutzer auswählt.
<i>Kursivschrift</i>	Parameter, die der Benutzer durch Namen oder Werte ersetzen muss, werden in Kursivschrift hervorgehoben.
Monospace-Schrift	Beispiele für bestimmte Datenwerte, Beispiele für Textanzeigen, Beispiele für Abschnitte von Programmcode, wie ihn ein Programmierer schreiben könnte, Systemnachrichten und Informationen, die Sie unverändert eingeben müssen, werden in Monospace-Schrift hervorgehoben.

Groß-/Kleinschreibung in AIX

Im Betriebssystem AIX wird grundsätzlich zwischen Groß- und Kleinbuchstaben unterschieden. Beispielsweise können Sie den Befehl **ls** zum Auflisten von Dateien verwenden. Wenn Sie **LS** eingeben, meldet das System, dass der Befehl nicht gefunden wurde. Ebenso sind **FILEA**, **FiLea** und **filea** in AIX drei unterschiedliche Dateien, die sich durchaus in einem Verzeichnis befinden können. Stellen Sie stets sicher, dass Sie die richtige Schreibweise verwenden, um unerwünschte Aktionen zu verhindern.

ISO 9000

Für die Entwicklung und Herstellung dieses Produkts wurden Qualitätssysteme gemäß ISO 9000 verwendet.

Verwaltung von Betriebssystem und Einheiten

Systemadministratoren und Systembenutzer können sich hier mit der Ausführung von Tasks, wie z. B. der Ausführung von Befehlen, der Verarbeitung von Prozessen, der Verarbeitung von Dateien und Verzeichnissen, dem Sichern und Wiederherstellen des Systems, der Verwaltung des physischen und logischen Speichers und den grundlegenden Druckaufgaben, vertraut machen.

Zu den weiteren hilfreichen Themen für Benutzer und Systemadministratoren, die in diesem Handbuch behandelt werden, gehören das Erstellen und das Festlegen der Größe eines Paging-Bereichs, das Verwalten des virtuellen Speichers, das Sichern und Wiederherstellen des Systems, das Verwalten von Hardware- und Pseudoeinheiten, das Verwenden des System Resource Controller (SRC), das Schützen von Dateien, das Verwenden von Speichermedien, das Anpassen von Umgebungsdateien und das Schreiben von Shell-Scripts. Sie finden diesen Abschnitt auch auf der zum Betriebssystem gelieferten Dokumentations-CD.

Neuerungen im Betriebssystem und bei der Einheitenverwaltung

In diesem Abschnitt finden Sie Informationen zu den neuen und signifikant geänderten Informationen in den Abschnitten zum Betriebssystem und zur Einheitenverwaltung.

Kennzeichnung neuer und geänderter Informationen

In dieser PDF-Datei sind neue und geänderte Informationen anhand der Änderungsmarkierungen (I) am linken Rand erkennbar.

Oktober 2017

Im Folgenden finden Sie eine Zusammenfassung der Aktualisierungen, die in dieser Themensammlung vorgenommen wurden:

- Der Abschnitt „Unterstützung der Speicherplatzrückforderung für logische Datenträger“ auf Seite 413 wurde hinzugefügt.
- Die Informationen zu den im Betriebssystem AIX unterstützten USB-Einheiten im Abschnitt „Unterstützung von USB-Einheiten“ auf Seite 613 wurden aktualisiert.

Juni 2017

Im Folgenden finden Sie eine Zusammenfassung der Aktualisierungen, die in dieser Themensammlung vorgenommen wurden:

- Im Abschnitt „Cachestatistiken überwachen“ auf Seite 623 wurden Informationen zur Statistiküberwachung hinzugefügt.
- Im Abschnitt „Verwaltung MPIO-fähiger Einheiten“ auf Seite 577 wurden Informationen zum Befehl `lsmpio` hinzugefügt.

Betriebssystemverwaltung

Sie können Befehle für die Verwaltung des Systemstarts und der Systemsicherung, des Systemabschlusses, der System-Shell und Shellumgebungen, der Systemressourcen und weiterer Teile von AIX verwenden.

Die Betriebssystemverwaltung ist die Aufgabe einer Einzelperson, die in der UNIX-Literatur gewöhnlich als Systemadministrator bezeichnet wird. Leider sind nur einige wenige Aktivitäten des Systemadministrators

rators unkompliziert genug, dass sie zu Recht Verwaltungsaufgaben genannt werden können. Dieses und andere zugehörige Handbücher sollen Systemadministratoren bei ihren zahlreichen Pflichten unterstützen.

Dieses Betriebssystem stellt eine eigene Version der Systemverwaltungsunterstützung bereit, um den Bedienungskomfort zu erhöhen und die Sicherheit und die Integrität zu verbessern.

Verfügbare Schnittstellen für die Systemverwaltung

Zusätzlich zur herkömmlichen Systemverwaltung über die Befehlszeile unterstützt dieses Betriebssystem die SMIT-Schnittstellen.

Im Folgenden werden die SMIT-Schnittstellen vorgestellt:

- System Management Interface Tool (SMIT) ist eine menübasierte Benutzerschnittstelle, die aus den Optionen, die Sie auswählen, Befehle erstellt und anschließend ausführt.

Mit SMIT können Sie die folgenden Aktionen ausführen:

- Software installieren, aktualisieren und verwalten
- Einheiten konfigurieren
- Plattenspeichereinheiten als Datenträgergruppen und logische Datenträger konfigurieren
- Dateisysteme und Paging-Bereich erstellen und erweitern
- Benutzer und Gruppen verwalten
- Netze und DFV-Anwendungen konfigurieren
- Drucken
- Fehlerbestimmung durchführen
- Jobs planen
- Systemressourcen und Workload verwalten
- Systemumgebungen verwalten
- Daten für Cluster-Systeme verwalten
- Eine objektorientierte grafische Benutzerschnittstelle, die dieselben Systemverwaltungstasks wie SMIT unterstützt, diese jedoch durch folgende Features vereinfacht:
 - Reduzierung der Benutzerfehler durch Fehlerprüfung und Dialogdesign
 - Schrittweises Vorgehen bei neuen oder komplexen Tasks
 - Erweiterte Optionen für erfahrenere Administratoren
 - Einfachere Visualisierung komplexer Daten oder Beziehungen zwischen Systemobjekten
 - Überwachung der Systemaktivität und Warnung des Administrators beim Eintreten vordefinierter Ereignisse
 - Bereitstellung von Kontexthilfen, Übersichten, Tipps und Links zur Onlinedokumentation

Elementare Softwaredaten

Bestimmte Informationen zu Softwareprodukten und ihren installierbaren Optionen werden in der Datenbank für elementare Softwaredaten (SWVPD, Software Vital Product Data) verwaltet.

Die elementaren Softwaredaten setzen sich aus einer Reihe von Befehlen und ODM-Objektklassen für die Verwaltung von Softwaredaten zusammen. Mit den bereitgestellten SWVPD-Befehlen kann der Benutzer installierte Softwareprodukte abfragen (**lsipp**) und überprüfen (**lppchk**). Die ODM-Objektklassen definieren den Umfang und das Format der Softwareprodukt Daten, die verwaltet werden.

Der Befehl **installp** verwendet den ODM, um die folgenden Informationen in der SWVPD-Datenbank zu verwalten:

- Name des installierten Softwareprodukts

- Version des Softwareprodukts
- Release-Level des Softwareprodukts, das anzeigt, dass an der externen Programmierschnittstelle des Softwareprodukts Änderungen vorgenommen wurden
- Modifikationsstufe des Softwareprodukts, die anzeigt, dass Änderungen vorgenommen wurden, die sich jedoch nicht auf die externe Programmierschnittstelle des Softwareprodukts auswirken
- Fixversion des Softwareprodukts, die anzeigt, dass kleinere Aktualisierungen vorgenommen wurden, die zu einem späteren Zeitpunkt in eine reguläre Modifikationsstufe integriert werden
- Feld für die Fix-ID
- Namen, Prüfsummen und Größen der Dateien, aus denen sich das Softwareprodukt bzw. die Option zusammensetzt
- Installationsstatus des Softwareprodukts: applying (wird angewendet), applied (angewendet), committing (wird festgeschrieben), committed (festgeschrieben), rejecting (wird zurückgewiesen) oder broken (fehlerhaft).

Betriebssystemaktualisierungen

Das Betriebssystempaket ist in Dateigruppen aufgeteilt. Jede Dateigruppe enthält eine Gruppe logisch zusammengehöriger, an den Kunden auslieferbare Dateien. Jede Dateigruppe kann einzeln installiert und aktualisiert werden.

Überarbeitungen von Dateigruppen werden mit den so genannten VRMF-Levels (Versions, Release, Maintenance and Fix) verfolgt. Nach Konvention wird nach jeder Aktualisierung einer AIX-Dateigruppe die Fixversion angepasst. Wird ein AIX-Wartungspaket oder Technology Level angewendet, wird die Modifikationsstufe angepasst und die Fixversion auf null zurückgesetzt. Die Erstinstallation einer Version von AIX, z. B. AIX 6.1, wird als Basisinstallation bezeichnet. Das Betriebssystem unterstützt Aktualisierungen seiner Features und Funktionen, die als Wartungspaket, Technology Level, PTF (Program Temporary Fix, vorläufige Programmkorrektur) oder Service-Pack (Gruppe von PTFs) gepackt sein können.

Wartungspakete und Technology Levels

Wartungspakete und Technology Levels enthalten neue Funktionen, mit denen das Release aktualisiert werden soll. In einem Wartungspaket wird der Teil Wartung (engl. Maintenance) von VRMF aktualisiert. Das erste Wartungspaket für AIX 6.1 ist beispielsweise 6.1.1.0, das zweite 6.1.2.0 usw. Zum Auflisten des Wartungspakets verwenden Sie den Befehl **oslevel -r**.

Geben Sie Folgendes ein, um das auf einem bestimmten System installierte Wartungspaket oder Technology Level zu ermitteln:

```
oslevel
```

Verwenden Sie den folgenden Befehl, um festzustellen, welche Dateigruppen aktualisiert werden müssen, damit das System ein bestimmtes Wartungspaket oder Technology Level (in diesem Beispiel 6.1.1.0) erreicht:

```
oslevel -l 6.1.1.0
```

Verwenden Sie den folgenden Befehl, um festzustellen, ob ein empfohlenes Wartungspaket oder Technology Level (in diesem Beispiel 6100-02) installiert ist:

```
oslevel -r 6100-02
```

Verwenden Sie den folgenden Befehl, um festzustellen, welche Dateigruppen aktualisiert werden müssen, damit das System das Wartungspaket oder Technology Level 6100-02 erreicht:

```
oslevel -rl 6100-02
```

Verwenden Sie den folgenden Befehl, um das Wartungspaket oder Technology Level einer bestimmten Dateigruppe (in diesem Beispiel bos.mp) zu bestimmen:

```
lslpp -L bos.mp
```

PTFs Zwischen Releases können Sie PTFs erhalten, mit denen ein bestimmtes Problem behoben oder verhindert wird. Eine bestimmte Installation kann einige, alle oder auch keinen der verfügbaren PTFs erfordern.

Empfohlene Wartungspakete (RMP, Recommended Maintenance Package)

Ein empfohlenes Wartungspaket ist eine Gruppe von PTFs zwischen Technology Levels, die ausführlich zusammen getestet wurden und für eine vorbeugende Wartung empfohlen werden.

Vorläufige Fixes

Ein vorläufiger Fix ist einem PTF sehr ähnlich, wird aber gewöhnlich angeboten, wenn kein PTF verfügbar ist. Vorläufige Fixes werden auch freigegeben, wenn der PTF ein System auf die nächste Wartungsstufe aktualisieren würde, Benutzer ihre Systeme aber lieber auf der aktuellen Stufe belassen möchten.

Wie Sie Version und Release-Level, Wartungspaket, Technology Level und Service-Pack-Level sowie die Dateigruppen ermitteln können, die aktualisiert werden müssen, um ein bestimmtes Level zu erreichen, finden Sie in den Abschnitten zu den Befehlen `oslevel` und `lspp` in der Veröffentlichung *Commands Reference*.

Systemstart

Wenn das Basisbetriebssystem gestartet wird, leitet das System eine komplexe Folge von Tasks ein. Unter normalen Bedingungen werden diese Tasks automatisch ausgeführt.

Es gibt verschiedene Situationen, in denen Sie das System anweisen möchten, einen Warmstart durchzuführen, z. B. um das System zu veranlassen, neu installierte Software zu erkennen, Peripheriegeräte zurückzusetzen, Routinewartungstasks wie die Überprüfung von Dateisystemen durchzuführen oder eine Wiederherstellung nach einer Systemblockierung oder einem Systemabsturz vorzunehmen. Informationen zu diesen Prozeduren finden Sie in den folgenden Abschnitten.

Zugehörige Tasks:

„Ein beschädigtes Boot-Image erneut erstellen“ auf Seite 37

Im Folgenden wird beschrieben, wie Sie ein beschädigtes Boot-Image identifizieren und erneut erstellen.

Systemstart verwalten

Es gibt mehrere Szenarios, in dem Sie Ihr System booten bzw. einen Warmstart durchführen müssen. Zum Herunterfahren Ihres Systems bzw. zum Durchführen eines Warmstarts können Sie den Befehl "shutdown" bzw. "reboot" verwenden. Sie sollten den Befehl "shutdown" verwenden, wenn mehrere Benutzer am System angemeldet sind.

Warmstart eines aktiven Systems durchführen:

Da Prozesse aktiv sein können, die kontrollierter beendet werden sollten, als es ein **Warmstart** zulässt, ist der Befehl **shutdown** die bevorzugte Methode für alle Systeme.

Es gibt zwei Methoden für das Herunterfahren und den Warmstart Ihres Systems: **shutdown** und **reboot**. Verwenden Sie immer die Methode **shutdown**, wenn mehrere Benutzer am System angemeldet sind.

Task	SMIT-Direktaufruf	Befehl oder Datei
Warmstart eines Mehrbenutzersystems	smit shutdown	shutdown -r
Warmstart eines Einzelbenutzersystems	smit shutdown	shutdown -r oder reboot

Warmstart eines nicht reagierenden Systems über Fernzugriff:

Die Funktion für fernen Warmstart ermöglicht einen Warmstart des Systems über einen nativen (integrierten) Systemanschluss.

Die integrierten *Systemanschlüsse* von POWER5-Systemen gleichen seriellen Anschlüssen. Die Systemanschlüsse sind jedoch nur für speziell unterstützte Funktionen verfügbar.

Es wird ein Warmstart des Systems durchgeführt, wenn die mit **reboot_string** angegebene Zeichenfolge für Warmstart am Anschluss empfangen wird. Diese Funktion ist hilfreich, wenn das System ansonsten nicht reagiert, aber noch in der Lage ist, Interrupts am Systemanschluss zu verarbeiten. Der ferne Warmstart kann jeweils nur an einem nativen Systemanschluss aktiviert werden. Benutzer müssen selbst für die externe Sicherheit an dem Anschluss sorgen. Diese Funktion wird mit der höchsten Interrupt-Klasse für Einheiten ausgeführt. Wenn der UART (Universal Asynchronous Receiver/Transmitter) den Übertragungspuffer nicht schnell genug löscht, kann dies dazu führen, dass Einheiten Daten verlieren, wenn ihre Puffer in dieser Zeit überlaufen. Es wird empfohlen, diese Funktion nur für den Warmstart einer Maschine zu verwenden, die anderweitig blockiert ist und an der keine ferne Anmeldung möglich ist. Dateisysteme werden *nicht* synchronisiert, und es besteht die Möglichkeit, dass einige Daten verloren gehen, die nicht auf die Platte zurückgeschrieben wurden. Wenn der ferne Warmstart aktiviert ist, wird dringend empfohlen, den Anschluss nicht für andere Zwecke zu verwenden, insbesondere für Dateiübertragungen, um einen versehentlichen Warmstart zu vermeiden.

Die Durchführung des fernen Warmstarts wird mit zwei Attributen des nativen Systemanschlusses gesteuert.

reboot_enable

Gibt an, ob an diesem Anschluss beim Empfang der fernen **Zeichenfolge für Warmstart** ein Warmstart durchgeführt wird, und wenn ja, ob vor dem Warmstart ein Systemspeicherauszug erstellt wird.

no - Gibt an, dass der ferne Warmstart inaktiviert ist.
reboot - Gibt an, dass der ferne Warmstart aktiviert ist.
dump - Gibt an, dass der ferne Warmstart aktiviert ist und dass vor dem Warmstart ein Systemspeicherauszug auf der primären Speicherauszugseinheit erstellt wird.

reboot_string

reboot_string gibt die Zeichenfolge für Warmstart an, auf die der serielle Anschluss achtet, wenn die Funktion für fernen Warmstart aktiviert ist. Wenn die Funktion für fernen Warmstart aktiviert ist, wird beim Empfang der **Zeichenfolge für Warmstart** am Anschluss das Zeichen > übertragen, und das System ist für den Warmstart bereit. Wenn das Zeichen 1 empfangen wird, wird ein Warmstart durchgeführt. Jedes andere Zeichen als 1 bricht den Warmstartprozess ab. Die **Zeichenfolge für Warmstart** darf maximal 16 Zeichen lang sein und keine Leerzeichen, Doppelpunkte, Gleichheitszeichen, Nullen, Zeilenvorschubzeichen und Zeichen vom Typ Strg-\ enthalten.

Der ferne Warmstart kann über SMIT oder die Befehlszeile aktiviert werden. Wenn Sie SMIT verwenden möchten, können Sie für ein konfiguriertes nicht grafikfähiges Terminal (TTY) den Pfad **Systemumgebungen -> Funktion für Verwaltung des fernen Warmstarts** verwenden. Konfigurieren Sie ein neues nicht grafikfähiges Terminal, kann der ferne Warmstart über die Menüs **TTY hinzufügen** oder **Merkmale eines TTY ändern/anzeigen** aktiviert werden. Der Zugriff über diese Menüs erfolgt über den Pfad **Einheiten -> TTY**.

In der Befehlszeile wird der Befehl **mkdev** oder **chdev** für die Aktivierung des fernen Warmstarts verwendet. Der folgende Befehl aktiviert beispielsweise den fernen Warmstart (mit der Speicherauszugsoption) und setzt **reboot_string**, d. h. die Zeichenfolge für Warmstart, für **tty1** auf **ReBoOtMe**.

```
chdev -l tty1 -a remreboot=dump -a reboot_string=ReBoOtMe
```

Dieser Beispielbefehl aktiviert den fernen Warmstart für **tty0** mit der aktuellen Einstellung von **reboot_string** lediglich in der Datenbank (er wird erst beim nächsten Warmstart wirksam).

```
chdev -P -l tty0 -a remreboot=reboot
```

Wenn das nicht grafikfähige Terminal als normaler Anschluss verwendet wird, müssen Sie den Befehl **pdisable** verwenden, bevor Sie den fernen Warmstart aktivieren. Mit dem Befehl **penable** können Sie den Anschluss nachher wieder aktivieren.

Zugehörige Informationen:

Function differences between system ports and serial ports

Zu Wartungszwecken von der Festplatte booten:

Sie können eine Maschine im Wartungsmodus von einer Festplatte booten.

Voraussetzungen

Es darf kein bootfähiger austauschbarer Datenträger (Band oder CD-ROM) im Laufwerk eingelegt sein. Spezielle Anweisungen zum Booten im Wartungsmodus für Ihr Modell können Sie der Hardwareokumentation entnehmen.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Maschine im Wartungsmodus von einer Festplatte zu booten:

1. Wenn Sie einen Warmstart durchführen, schalten Sie die Maschine aus und wieder ein, oder drücken Sie den Grundstellungsknopf.
2. Drücken Sie die Tastenkombination für einen Warmstart im Wartungsmodus, die in Ihrer Hardwareokumentation beschrieben ist.
3. Die Maschine wird bis zu einem Punkt gebootet, an dem eine Konsoleinheit verfügbar ist. Wenn ein Systemspeicherauszug vorhanden ist, der abgerufen werden muss, wird in der Konsole das Menü für Systemspeicherauszüge angezeigt.

Anmerkung:

- a. Falls die Konfiguration der Konsole scheitert und ein abzurufender Speicherauszug vorhanden ist, blockiert das System. Das System muss von einem austauschbaren Datenträger gebootet werden, damit der Speicherauszug abgerufen werden kann.
 - b. Das System legt den Speicherauszug automatisch auf der angegebenen Speicherauszugseinheit ab, wenn der Grundstellungsknopf gedrückt wird. Wenn Sie die Definition der primären oder sekundären Speicherauszugseinheit in einem aktiven System ändern möchten, finden Sie diesbezügliche Informationen in der Beschreibung des Befehls **sysdumpdev**.
4. Wenn kein Systemspeicherauszug vorhanden ist oder wenn der Systemspeicherauszug kopiert wurde, werden die Bedienungsanweisungen für die Diagnose angezeigt. Drücken Sie die Eingabetaste, um das Menü **Funktionsauswahl** aufzurufen.
 5. Im Menü **Funktionsauswahl** können Sie zwischen dem Diagnose- und dem Einzelbenutzermodus wählen.

Einzelbenutzermodus: Wenn Sie in einer Einzelbenutzerumgebung Wartungsarbeiten vornehmen möchten, müssen Sie diese Option (Option 5) auswählen. Das System setzt den Bootvorgang fort und wechselt in den Einzelbenutzermodus. Wartungsarbeiten, die voraussetzen, dass das System im eigenständigen Modus ausgeführt wird, können in diesem Modus ausgeführt werden. Außerdem kann bei Bedarf der Befehl **bosboot** ausgeführt werden.

Zugehörige Informationen:

Starting a System Dump

Ein abgestürztes System booten:

Es kann vorkommen, dass Sie ein System booten müssen, das ohne ordnungsgemäß durchgeführten Systemabschluss gestoppt wurde (Absturz).

Die Voraussetzungen für diese Prozedur sind im Folgenden beschrieben:

- Ihr System ist aufgrund ungewöhnlicher Bedingungen abgestürzt und wurde nicht ordnungsgemäß heruntergefahren.
- Ihr System ist ausgeschaltet.

Diese Prozedur beschreibt die grundlegenden Schritte, die ausgeführt werden müssen, wenn ein automatischer Wiederanlauf des Systems nach einem Absturz nicht möglich ist. Führen Sie die folgenden Schritte aus:

1. Vergewissern Sie sich, dass alle Hardware- und Peripherieeinheiten ordnungsgemäß angeschlossen sind.
2. Schalten Sie alle Peripherieeinheiten ein.
3. Beobachten Sie, ob am Bildschirm Informationen zur automatischen Hardware diagnose angezeigt werden.
 - a. Wenn die Hardware diagnostetests nicht erfolgreich sind, ziehen Sie die Hardware dokumentation zu Rate.
 - b. Wenn alle Hardware diagnostetests erfolgreich sind, schalten Sie die Systemeinheit ein.

Unbekanntes Rootkennwort zurücksetzen:

Die folgende Prozedur beschreibt, wie Sie die Rootzugriffsrechte wiederherstellen, wenn das Rootkennwort des Systems nicht verfügbar oder unbekannt ist.

Für die folgenden Schritte muss das System heruntergefahren werden. Sie sollten für diese Prozedur einen Zeitpunkt wählen, der so wenig wie möglich Auswirkungen auf Ihren Arbeitsablauf hat.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Legen Sie den Produktdatenträger für die Version und Stufe in das Laufwerk ein, die derzeit installiert ist.
2. Schalten Sie die Maschine ein.
3. Wenn die Anzeige mit den Symbolen erscheint oder wenn Sie einen doppelten Signalton hören, drücken Sie die Taste F1 so oft, bis das Menü **System Management Services** erscheint.
4. Wählen Sie **Multiboot** aus.
5. Wählen Sie **Installieren von** aus.
6. Wählen Sie die Einheit mit dem eingelegten Produktdatenträger aus, und wählen Sie dann **Installieren** aus.
7. Wählen Sie das AIX-Versionssymbol aus.
8. Definieren Sie Ihr aktuelles System als Systemkonsole. Drücken Sie dazu die Taste F1 und anschließend die Eingabetaste.
9. Wählen Sie die Nummer der von Ihnen bevorzugten Sprache aus, und drücken Sie anschließend die Eingabetaste.
10. Wählen Sie **Wartungsmodus für Wiederherstellung des Systembetriebs starten** aus. Geben Sie dazu eine 3 ein, und drücken Sie anschließend die Eingabetaste.
11. Wählen Sie die Option **Auf eine Stammdatenträgergruppe (rootvg) zugreifen** aus. Daraufhin erscheint eine Nachricht, die Ihnen mitteilt, dass Sie ohne erneutes Booten des Systems nicht in die Installationsmenüs zurückkehren können, falls Sie die Stammdatenträgergruppe an diesem Punkt ändern.
12. Geben Sie 0 ein, und drücken Sie die Eingabetaste.
13. Geben Sie die Nummer der gewünschten Datenträgergruppe aus der Liste ein, und drücken Sie anschließend die Eingabetaste.
14. Wählen Sie **Auf diese Datenträgergruppe zugreifen und Shell starten** aus. Geben Sie dazu eine 1 ein, und drücken Sie anschließend die Eingabetaste.
15. Geben Sie an der Eingabeaufforderung mit # (dem engl. Nummernzeichen) den Befehl **passwd** ein, um das Rootkennwort zurückzusetzen. Beispiel:

```
# passwd
Kennwort für "root" wird geändert
Neues Kennwort für root:
Neues Kennwort erneut eingeben:
```

16. Geben Sie Folgendes ein, um alle Daten aus dem Puffer auf die Festplatte zu schreiben und das System erneut zu booten:

```
sync;sync;sync;reboot
```

Wenn die Anmeldeanzeige erscheint, sollten Sie durch Eingabe des in Schritt 15 festgelegten Kennworts wieder Rootzugriffsrechte haben.

Zugehörige Informationen:

passwd command

reboot command

Systeme mit integriertem Grafiksubsystem booten:

Wenn die Maschine nur mit einem integrierten Grafiksubsystem installiert wurde und dem System später ein weiterer Grafikadapter hinzugefügt wird, geschieht Folgendes:

1. Ein neuer Grafikadapter wird dem System hinzugefügt, und die zugehörige Einheits-treiber-Software wird installiert.
2. Es wird ein Warmstart des Systems durchgeführt, und eines der folgenden Ereignisse tritt ein:
 - a. Wenn die Systemkonsole als `/dev/lft0` definiert ist (der Befehl **lscons** zeigt diese Informationen an), wird der Benutzer aufgefordert, den Bildschirm auszuwählen, der bei einem Warmstart als Systemkonsole verwendet werden soll. Wählt der Benutzer einen Grafikadapter aus (kein nicht grafikfähiges Terminal), wird dieser Adapter gleichzeitig der neue Standardbildschirm. Wählt der Benutzer anstelle einer LFT-Einheit ein nicht grafikfähiges Terminal aus, erscheint keine Systemanmeldung. Führen Sie erneut einen Warmstart durch, und der Anmeldebildschirm erscheint auf dem nicht grafikfähigen Terminal. Es wird vorausgesetzt, dass der Benutzer nicht das nicht grafikfähige Terminal als Systemkonsole auswählt, wenn er einen zusätzlichen Grafikadapter zum System hinzufügt und die Systemkonsole eine LFT-Einheit ist.
 - b. Wenn als Systemkonsole ein nicht grafikfähiges Terminal definiert wird, wird der neu hinzugefügte Grafikadapter bei einem Warmstart zum Standardbildschirm.

Anmerkung: Da das nicht grafikfähige Terminal die Systemkonsole ist, bleibt es auch die Systemkonsole.

3. Wenn `/dev/lft0` als Systemkonsole definiert ist, wird nach einem Warmstart DPMS inaktiviert, und der Auswahltext für die Systemkonsole wird für unbegrenzte Zeit am Bildschirm angezeigt. Führen Sie einen erneuten Warmstart durch, um DPMS erneut zu aktivieren.

Ausführung von Scripts für Ausführungsebenen implementieren:

Mit Scripts für Ausführungsebenen können Benutzer ausgewählte Anwendungen starten und stoppen, während die Ausführungsebene geändert wird.

Speichern Sie Scripts für Ausführungsebenen im Unterverzeichnis von `/etc/rc.d`, das für die jeweilige Ausführungsebene spezifisch ist:

- `/etc/rc.d/rc2.d`
- `/etc/rc.d/rc3.d`
- `/etc/rc.d/rc4.d`
- `/etc/rc.d/rc5.d`
- `/etc/rc.d/rc6.d`
- `/etc/rc.d/rc7.d`
- `/etc/rc.d/rc8.d`

- /etc/rc.d/rc9.d

/etc/rc.d/rc führt die Scripts im angegebenen Verzeichnis aus, wenn sich die Ausführungsebene ändert. Dabei werden zuerst die Scripts zum Stoppen der Anwendung und anschließend die Scripts zum Starten der Anwendung ausgeführt.

Anmerkung: Scripts, die mit K beginnen, sind Stoppscripts, und Scripts, die mit S beginnen, sind Startscripts.

Datei /etc/inittab ändern:

Es stehen vier Befehle zur Verfügung, mit denen die Datensätze in der Datei etc/inittab geändert werden können.

Datensätze hinzufügen - Befehl mkitab

Wenn Sie einen Datensatz zur Datei /etc/inittab hinzufügen möchten, geben Sie Folgendes an einer Eingabeaufforderung ein:

```
mkinitab ID:Ausführungsebene:Aktion:Befehl
```

Wenn Sie beispielsweise einen Datensatz für tty2 hinzufügen möchten, geben Sie Folgendes an einer Eingabeaufforderung ein:

```
mkinitab tty002:2:respawn:/usr/sbin/getty /dev/tty2
```

Erklärungen zum Beispiel:

Eintrag	Beschreibung
tty002	Gibt das Objekt an, dessen Ausführungsebene Sie definieren.
2	Gibt die Ausführungsebene an, auf der dieser Prozess ausgeführt wird.
respawn	Gibt die Aktion an, die der Befehl init für diesen Prozess ausführen soll.
/usr/sbin/getty /dev/tty2	Gibt den auszuführenden Shellbefehl an.

Datensätze ändern - Befehl chitab

Wenn Sie einen Datensatz in der Datei /etc/inittab ändern möchten, geben Sie Folgendes an einer Eingabeaufforderung ein:

```
chitab ID:Ausführungsebene:Aktion:Befehl
```

Wenn Sie beispielsweise einen Datensatz für tty2 ändern möchten, so dass dieser Prozess auf den Ausführungsebenen 2 und 3 ausgeführt wird, geben Sie Folgendes ein:

```
chitab tty002:23:respawn:/usr/sbin/getty /dev/tty2
```

Erklärungen zum Beispiel:

Eintrag	Beschreibung
tty002	Gibt das Objekt an, dessen Ausführungsebene Sie definieren.
23	Gibt die Ausführungsebenen an, auf denen dieser Prozess ausgeführt wird.
respawn	Gibt die Aktion an, die der Befehl init für diesen Prozess ausführen soll.
/usr/sbin/getty /dev/tty2	Gibt den auszuführenden Shellbefehl an.

Datensätze auflisten - Befehl lsitab

Wenn Sie alle Datensätze in der Datei /etc/inittab auflisten möchten, geben Sie Folgendes an einer Eingabeaufforderung ein:

```
lsitab -a
```

Wenn Sie einen bestimmten Datensatz in der Datei `/etc/inittab` auflisten möchten, geben Sie Folgendes ein:

`lsitab ID`

Wenn Sie beispielsweise den Datensatz für `tty2` auflisten möchten, geben Sie Folgendes ein: `lsitab tty2`.

Datensätze entfernen - Befehl `rmitab`

Wenn Sie einen Datensatz aus der Datei `/etc/inittab` entfernen möchten, geben Sie Folgendes an einer Eingabeaufforderung ein:

`rmitab ID`

Wenn Sie beispielsweise den Datensatz für `tty2` entfernen möchten, geben Sie Folgendes ein: `rmitab tty2`.

Zugehörige Konzepte:

„Systemausführungsebene“ auf Seite 15

Die Systemausführungsebene gibt den Systemstatus an und definiert, welche Prozesse gestartet werden.

Reaktivierung eines inaktiven Systems:

Gründe für die Inaktivität Ihres Systems können Hardwareprobleme, Softwareprobleme oder eine Kombination von beiden sein.

Diese Prozedur beschreibt detailliert, wie Sie das Problem beheben und das System erneut starten. Falls Ihr System nach Abschluss dieser Prozedur weiterhin inaktiv ist, ziehen Sie die Informationen zur Fehlerbestimmung in Ihrer Hardwaredokumentation zu Rate.

Verwenden Sie die folgenden Prozeduren, um ein inaktives System zu reaktivieren:

Hardwareüberprüfung:

Es gibt verschiedene Methoden für die Überprüfung Ihrer Hardware.

Sie können Ihre Hardware mit den folgenden Methoden überprüfen:

Stromversorgung überprüfen:

Wenn die Betriebsanzeige an Ihrem System leuchtet, fahren Sie mit dem Abschnitt **Anzeige an der Bedienerkonsole überprüfen** fort.

Wenn die Betriebsanzeige an Ihrem System nicht leuchtet, stellen Sie sicher, dass das System eingeschaltet ist und der Netzstecker eingesteckt ist.

Anzeige an der Bedienerkonsole überprüfen:

Wenn Ihr System eine Bedienerkonsolanzeige besitzt, prüfen Sie, ob dort Nachrichten angezeigt werden.

Wenn die Anzeige der Bedienerkonsole Ihres Systems leer ist, fahren Sie mit dem Abschnitt **Bildschirm oder Terminal aktivieren** fort.

Wenn die Anzeige der Bedienerkonsole Ihres Systems nicht leer ist, suchen Sie im Servicehandbuch zu Ihrer Einheit nach Informationen zu den Ziffern, die in der Bedienerkonsolanzeige erscheinen.

Bildschirm oder Terminal aktivieren:

Überprüfen Sie wie folgt verschiedene Komponenten Ihres Bildschirms bzw. Terminals:

- Stellen Sie sicher, dass das Bildschirmkabel fest an den Bildschirm und die Systemeinheit angeschlossen ist.
- Stellen Sie sicher, dass das Tastaturkabel fest angeschlossen ist.
- Stellen Sie sicher, dass das Mauskabel fest angeschlossen ist.
- Stellen Sie sicher, dass der Bildschirm eingeschaltet ist und dass die Betriebsanzeige leuchtet.
- Stellen Sie die Helligkeitsregler am Bildschirm ein.
- Stellen Sie sicher, dass die Übertragungseinstellungen des Terminals korrekt sind.

Wenn Ihr System jetzt aktiv ist, haben Sie das Problem durch Ihre Hardwareüberprüfungen behoben.

Zugehörige Tasks:

„System erneut starten“ auf Seite 13

Zusätzlich zur Überprüfung der Hardware und Überprüfung der Prozesse können Sie Ihr System erneut starten, um ein inaktives System zu reaktivieren.

„Prozesse überprüfen“

Ein gestoppter oder blockierter Prozess kann dazu führen, dass Ihr System inaktiv wird.

Prozesse überprüfen:

Ein gestoppter oder blockierter Prozess kann dazu führen, dass Ihr System inaktiv wird.

Sie können Ihre Systemprozesse mit den folgenden Methoden überprüfen:

1. Zeilenweises Verschieben erneut starten
2. Tastenkombination Strg+D verwenden
3. Tastenkombination Strg+C verwenden
4. Über ein fernes Terminal oder einen fernen Host anmelden
5. Blockierte Prozesse über Remote-Zugriff beenden

Zeilenweises Verschieben erneut starten:

Sie können das zeilenweise Verschieben, das mit der Tastenkombination Strg-S angehalten wurde, wie folgt erneut starten:

1. Aktivieren Sie das Fenster oder die Shell mit dem fehlerhaften Prozess.
2. Drücken Sie die Tastenkombination Strg-Q, um das zeilenweise Verschieben erneut zu starten. Mit der Tastenkombination Strg-S wird das zeilenweise Verschieben gestoppt und mit der Tastenkombination Strg-Q erneut gestartet.

Wenn diese Aktion das Problem mit dem inaktiven System nicht behebt, fahren Sie mit dem nächsten Abschnitt, **Tastenkombination Strg-D verwenden**, fort.

Tastenkombination Strg-D verwenden:

1. Aktivieren Sie das Fenster oder die Shell mit dem fehlerhaften Prozess.
2. Drücken Sie die Tastenkombination Strg-D. Mit der Tastenkombination Strg-D wird ein Dateiendesignal (EOF, End of File) an den Prozess gesendet. Die Tastenkombination Strg-D kann das Fenster bzw. die Shell schließen und Sie abmelden.

Wenn diese Aktion das Problem mit dem inaktiven System nicht behebt, fahren Sie mit dem nächsten Abschnitt, **Tastenkombination Strg-C verwenden**, fort.

Tastenkombination Strg-C verwenden:

Gehen Sie wie folgt vor, um einen gestoppten Prozess zu beenden:

1. Aktivieren Sie das Fenster oder die Shell mit dem fehlerhaften Prozess.

2. Drücken Sie die Tastenkombination Strg-C. Die Tastenkombination Strg-C stoppt die aktuelle Suche bzw. den aktuellen Filter.

Wenn diese Aktion das Problem mit dem inaktiven System nicht behebt, fahren Sie mit dem nächsten Abschnitt, **Über ein fernes Terminal oder einen fernen Host anmelden**, fort.

Über ein fernes Terminal oder einen fernen Host anmelden:

Melden Sie sich auf eine der folgenden beiden Arten über Remote-Zugriff an:

- Melden Sie sich über ein anderes Terminal an, wenn mehrere Terminals an Ihr System angeschlossen sind.
- Melden Sie sich über einen anderen Host im Netz an (falls Ihr System an ein Netz angeschlossen ist). Geben Sie dazu den Befehl **tn** wie folgt ein:

```
tn Name_Ihres_Systems
```

Das System fordert Sie zur Eingabe Ihres regulären Anmeldenamens und Ihres Kennworts auf, wenn Sie den Befehl **tn** verwenden.

Wenn Sie sich über ein fernes Terminal oder einen fernen Host am System anmelden können, fahren Sie mit dem nächsten Abschnitt, **Blockierte Prozesse über Remote-Zugriff beenden**, fort.

Wenn Sie sich nicht über ein fernes Terminal oder einen fernen Host am System anmelden können, müssen Sie das System erneut starten.

Sie können auch einen Systemspeicherauszug starten, um festzustellen, warum Ihr System inaktiv ist.

Blockierte Prozesse über Remote-Zugriff beenden:

Gehen Sie wie folgt vor, um einen blockierten Prozess über ein fernes Terminal zu beenden:

1. Listen Sie die aktiven Prozesse mit dem folgenden Befehl **ps** auf:

```
ps -ef
```

Die Flags **-e** und **-f** ermitteln alle aktiven und inaktiven Prozesse.

2. Ermitteln Sie die Prozess-ID des blockierten Prozesses.

Wenn Sie Hilfe zum Ermitteln von Prozessen benötigen, verwenden Sie den Befehl **grep** mit einem Suchbegriff. Zum Beenden des Prozesses **xlock** geben Sie beispielsweise Folgendes ein, um die Prozess-ID zu ermitteln:

```
ps -ef | grep xlock
```

Mit dem Befehl **grep** können Sie die Ausgabe des Befehls **ps** durchsuchen, um die Prozess-ID eines bestimmten Prozesses zu ermitteln.

3. Beenden Sie den Prozess mit dem folgenden Befehl **kill**:

Anmerkung: Wenn Sie Prozesse beenden möchten, die Sie nicht selbst eingeleitet haben, müssen Sie für die Verwendung des Befehls **kill** Rootberechtigung besitzen.

```
kill -9 Prozess-ID
```

Wenn Sie den fehlerhaften Prozess nicht ermitteln können, ist möglicherweise der zuletzt aktivierte Prozess die Ursache für Ihr inaktives System. Beenden Sie diesen Prozess, wenn Sie diesen für den Problemverursacher halten.

Wenn Sie mit Ihren Prozessüberprüfungen das Problem mit Ihrem inaktiven System nicht beheben können, müssen Sie das System erneut starten.

Zugehörige Konzepte:

„Hardwareüberprüfung“ auf Seite 10

Es gibt verschiedene Methoden für die Überprüfung Ihrer Hardware.

Zugehörige Tasks:

„System erneut starten“

Zusätzlich zur Überprüfung der Hardware und Überprüfung der Prozesse können Sie Ihr System erneut starten, um ein inaktives System zu reaktivieren.

Zugehörige Informationen:

System Dump Facility

System erneut starten:

Zusätzlich zur Überprüfung der Hardware und Überprüfung der Prozesse können Sie Ihr System erneut starten, um ein inaktives System zu reaktivieren.

Wenn Sie mit den in den Abschnitten „Hardwareüberprüfung“ auf Seite 10 und „Prozesse überprüfen“ auf Seite 11 beschriebenen Prozeduren das Problem, das für die Inaktivität Ihres Systems verantwortlich ist, nicht beheben können, müssen Sie das System erneut starten.

Anmerkung: Erstellen Sie vor dem Neustart des Systems einen Systemspeicherauszug.

1. Überprüfen Sie den Status der Booteinheit.

Ihr System wird entweder über einen austauschbaren Datenträger, eine externe Einheit, eine SCSI-Einheit (Small Computer System Interface), eine IDE-Einheit (Integrated Device Electronics) oder ein lokales Netz (LAN) gebootet. Entscheiden Sie, welche Methode für Ihr System angebracht ist, und verwenden Sie die folgenden Anweisungen zum Überprüfen der Booteinheit:

- Wenn Sie einen austauschbaren Datenträger, z. B. ein Band, verwenden, müssen Sie sicherstellen, dass der Datenträger ordnungsgemäß in das Laufwerk eingelegt ist.
- Wenn Sie eine IDE-Einheit verwenden, überprüfen Sie, ob die ID-Einstellungen der IDE-Einheit für die Adapter eindeutig sind. Wenn nur eine Einheit an den Adapter angeschlossen ist, muss die ID der IDE-Einheit auf die Mastereinheit eingestellt sein.
- Wenn Sie eine extern angeschlossene Einheit, z. B. ein Bandlaufwerk, verwenden, müssen Sie Folgendes sicherstellen:
 - Die Stromversorgung der Einheit ist aktiviert.
 - Die Einheitenkabel sind ordnungsgemäß an die Einheit und an die Systemeinheit angeschlossen.
 - Die Bereitschaftsanzeige leuchtet (sofern die Einheit eine solche besitzt).
- Wenn Sie eine externe SCSI-Einheit verwenden, überprüfen Sie, ob die Einstellungen für die SCSI-Adressen eindeutig sind.
- Wenn Sie ein LAN verwenden, stellen Sie sicher, dass das Netz aktiv und funktionsfähig ist.

Wenn die Booteinheit ordnungsgemäß arbeitet, fahren Sie mit dem nächsten Schritt fort.

2. Laden Sie Ihr Betriebssystem wie folgt:

- a. Schalten Sie die Stromversorgung des Systems aus.
- b. Warten Sie eine Minute.
- c. Schalten Sie die Stromversorgung des Systems ein.
- d. Warten Sie, ob das System bootet.

Wenn das Betriebssystem nicht geladen werden konnte, booten Sie die Festplatte im Wartungs- oder Hardware Diagnosemodus.

Wenn Sie das System auch auf diese Weise nicht erneut starten können, melden Sie das Problem mit Ihrem inaktiven System in einem SRN Ihrem Kundendienst.

Zugehörige Konzepte:

„Hardwareüberprüfung“ auf Seite 10

Es gibt verschiedene Methoden für die Überprüfung Ihrer Hardware.

Zugehörige Tasks:

„Prozesse überprüfen“ auf Seite 11

Ein gestoppter oder blockierter Prozess kann dazu führen, dass Ihr System inaktiv wird.

Zugehörige Informationen:

System Dump Facility

Boot-Images erstellen

Sie benötigen ein Boot-Image, um das Basisbetriebssystem zu installieren oder auf ein System zuzugreifen, das nicht vom Systemfestplattenlaufwerk bootet. Diese Prozedur beschreibt, wie Boot-Images erstellt werden. Das Boot-Image ist für jeden Einheitentyp verschieden.

Bei der Erstinstallation des Systems erstellt der Befehl **bosboot** aus dem Image eines RAM-Disk-Dateisystems (Random Access Memory, Arbeitsspeicher) und dem Betriebssystemkernel ein Boot-Image. Das Boot-Image wird auf einen bestimmten Datenträger, z. B. die Festplatte, übertragen. Beim Warmstart der Maschine wird das Boot-Image von dem Datenträger in den Hauptspeicher geladen. Weitere Informationen finden Sie in der Beschreibung des Befehls **bosboot** in der Veröffentlichung "Commands Reference".

Das zugeordnete RAM-Plattendateisystem enthält Konfigurationsroutinen für die folgenden Einheiten:

- **Platte**
- **Band**
- **CD-ROM**
- **Token-Ring-, Ethernet- oder FDDI-Einheit**
- Zur Verwendung des Befehls **bosboot** müssen Sie Rootberechtigung besitzen.
- Im Dateisystem /tmp müssen mindestens 20 MB freier Speicherplatz sein.
- Die physische Platte muss den logischen Bootdatenträger enthalten. Geben Sie an einer Eingabeaufforderung Folgendes ein, um festzustellen, welche Platteneinheit Sie angeben müssen:

```
lsvg -l rootvg
```

Der Befehl **lsvg -l** listet die logischen Datenträger in der Stammdatenträgergruppe (rootvg) auf. In dieser Liste finden Sie den Namen des logischen Bootdatenträgers.

Geben Sie anschließend an einer Eingabeaufforderung Folgendes ein:

```
lsvg -M rootvg
```

Der Befehl **lsvg -M** listet die physischen Platten mit den verschiedenen logischen Datenträgern auf.

Ein Boot-Image auf einem logischen Bootdatenträger erstellen:

Bei der Installation des Basisbetriebssystems (Neuinstallation oder Aktualisierung) wird der Befehl **bosboot** aufgerufen, der das Boot-Image auf den logischen Bootdatenträger kopiert. Der logische Bootdatenträger ist ein physisch zusammenhängender Bereich auf der Platte, der während der Installation mit dem Logical Volume Manager (LVM) erstellt wird.

Eine Liste der Voraussetzungen für diese Prozedur finden Sie im Abschnitt „Boot-Images erstellen“.

Der Befehl **bosboot** führt die folgenden Aktionen aus:

1. Er überprüft das Dateisystem, um festzustellen, ob genügend Platz verfügbar ist, um das Boot-Image zu erstellen.
2. Er erstellt mit dem Befehl **mkfs** und einer Prototypdatei ein RAM-Dateisystem.
3. Er ruft den Befehl **mkboot** auf, der den Kernel und das RAM-Dateisystem zu einem Boot-Image zusammenführt.

4. Er schreibt das Boot-Image auf den logischen Bootdatenträger.

Wenn Sie ein Boot-Image auf dem logischen Standardbootdatenträger auf der Festplatte erstellen möchten, geben Sie an einer Eingabeaufforderung Folgendes ein:

```
bosboot -a
```

ODER:

```
bosboot -ad /dev/ipldevice
```

Anmerkung: Führen Sie keinen Warmstart durch, wenn der Befehl **bosboot** beim Erstellen eines Boot-Image fehlschlägt. Beheben Sie das Problem, und führen Sie den Befehl **bosboot** erneut aus.

Sie müssen einen Warmstart durchführen, damit das neue Boot-Image verwendet werden kann.

Boot-Images für Netzeinheiten erstellen:

Sie können Boot-Images für einen Ethernet- oder einen Token-Ring-Bootvorgang erstellen.

Eine Liste der Voraussetzungen für diese Prozedur finden Sie im Abschnitt „Boot-Images erstellen“ auf Seite 14.

Geben Sie an einer Eingabeaufforderung Folgendes ein, um ein Boot-Image für einen Ethernet-Bootvorgang zu erstellen:

```
bosboot -ad /dev/ent
```

Geben Sie Folgendes ein, um ein Boot-Image für einen Token-Ring-Bootvorgang zu erstellen:

```
bosboot -ad /dev/tok
```

Systemausführungsebene

Die Systemausführungsebene gibt den Systemstatus an und definiert, welche Prozesse gestartet werden.

Wenn 3 als Systemausführungsebene eingestellt ist, werden alle Prozesse gestartet, die für diese Ausführungsebene definiert sind. Gegen Ende der Systembootphase des Bootprozesses wird die Ausführungsebene aus dem `initdefault`-Eintrag in der Datei `/etc/inittab` gelesen. Das System arbeitet so lange auf dieser Ausführungsebene, bis es ein Signal zum Ändern der Ausführungsebene empfängt. Die Systemausführungsebene kann mit dem Befehl `init` geändert werden. Die Datei `/etc/inittab` enthält einen Datensatz für jeden Prozess, der die Ausführungsebenen für diesen Prozess definiert. Wenn das System bootet, liest der Befehl `init` die Datei `/etc/inittab`, um festzustellen, welche Prozesse gestartet werden müssen.

Im Folgenden sind die derzeit definierten Ausführungsebenen aufgelistet:

Eintrag	Beschreibung
0-9	Wenn der Befehl <code>init</code> eine Umstellung auf eine der Ausführungsebenen 0 bis 9 durchführt, beendet er alle Prozesse der aktuellen Ausführungsebene und startet anschließend alle Prozesse erneut, die der neuen Ausführungsebene zugeordnet sind.
0-1	Reserviert für spätere Verwendung durch das Betriebssystem.
2	Standardausführungsebene.
3-9	Kann gemäß den Einstellungen des Benutzers definiert werden.
a, b, c	Wenn der Befehl <code>init</code> eine Änderung auf die Ausführungsebene <code>a</code> , <code>b</code> oder <code>c</code> anfordert, beendet er keine Prozesse der aktuellen Ausführungsebene, sondern startet einfach alle Prozesse erneut, die der neuen Ausführungsebene zugeordnet sind.
Q, q	Weist den Befehl <code>init</code> an, die Datei <code>/etc/inittab</code> erneut einzulesen.

Zugehörige Tasks:

„Datei `/etc/inittab` ändern“ auf Seite 9

Es stehen vier Befehle zur Verfügung, mit denen die Datensätze in der Datei `etc/inittab` geändert wer-

den können.

Systemausführungsebene ermitteln:

Bevor Sie Wartungsarbeiten am Betriebssystem vornehmen oder die Systemausführungsebene ändern, müssen Sie möglicherweise die verschiedenen Ausführungsebenen untersuchen.

Diese Prozedur beschreibt, wie Sie die Ausführungsebene ermitteln, auf der das System arbeitet, und wie Sie ein Protokoll früherer Ausführungsebenen anzeigen. Der Befehl **init** bestimmt die Systemausführungsebene.

Ermittlung der aktuellen Ausführungsebene

Geben Sie in der Befehlszeile `cat /etc/.init.state` ein. Das System zeigt eine Ziffer an. Dies ist die aktuelle Ausführungsebene. Weitere Informationen zu Ausführungsebenen finden Sie im Abschnitt zum Befehl **init** und im Abschnitt zur Datei `/etc/inittab`.

Ein Protokoll früherer Ausführungsebenen anzeigen:

Mit dem Befehl **fwtmp** können Sie ein Protokoll früherer Ausführungsebenen anzeigen.

Anmerkung: Zur Verwendung dieses Befehls muss der Code `bosext2.acct.obj` auf dem System installiert sein.

1. Melden Sie sich als Root an.
2. Geben Sie an einer Eingabeaufforderung Folgendes ein:

```
/usr/lib/acct/fwtmp </var/adm/wtmp |grep run-level
```

Das System zeigt Informationen ähnlich den folgenden an:

```
run-level 2 0 1 0062 0123 697081013 Sun Feb 2 19:36:53 CST 1992
run-level 2 0 1 0062 0123 697092441 Son Feb 2 22:47:21 CST 1992
run-level 4 0 1 0062 0123 698180044 Sam Feb 15 12:54:04 CST 1992
run-level 2 0 1 0062 0123 698959131 Son Feb 16 10:52:11 CST 1992
run-level 5 0 1 0062 0123 698967773 Mon Feb 24 15:42:53 CST 1992
```

Ausführungsebenen auf Mehrbenutzersystemen konfigurieren:

Sie können die Ausführungsebenen auf Mehrbenutzersystemen ändern.

1. Überprüfen Sie die Datei `/etc/inittab`, um sicherzustellen, dass die Ausführungsebene, auf die Sie wechseln möchten, die Prozesse unterstützt, die Sie ausführen. Der Prozess `getty` ist besonders wichtig, da er den Zugriff auf die Terminalleitung für die Systemkonsole und andere Anmeldungen steuert. Vergewissern Sie sich, dass der Prozess `getty` auf allen Ausführungsebenen aktiviert ist.
2. Verwenden Sie den Befehl **wall**, um alle Benutzer darüber zu informieren, dass Sie beabsichtigen, die Ausführungsebene zu ändern, und dass die Benutzer sich vom System abmelden sollen. Weitere Informationen finden Sie in der Beschreibung des Befehls **wall** in der Veröffentlichung "Commands Reference".
3. Verwenden Sie den Direktaufruf **smit telinit**, um auf das Menü **Systemausführungsebene festlegen** zuzugreifen.
4. Geben Sie im Feld **Systemausführungsebene** die neue Ausführungsebene ein.
5. Drücken Sie die Eingabetaste, um alle Einstellungen in dieser Prozedur zu implementieren. Das System antwortet, indem es Ihnen mitteilt, welche Prozesse aufgrund der geänderten Ausführungsebene beendet oder gestartet werden, und indem es die folgende Nachricht anzeigt:

```
INIT: New run level: n
```

n steht für die Nummer der neuen Ausführungsebene.

Ausführungsebenen auf Einzelbenutzersystemen konfigurieren:

Sie können die Ausführungsebenen auf Einzelbenutzersystemen ändern.

1. Überprüfen Sie die Datei `/etc/inittab`, um sicherzustellen, dass die Ausführungsebene, auf die Sie wechseln möchten, die Prozesse unterstützt, die Sie ausführen. Der Prozess `getty` ist besonders wichtig, da er den Zugriff auf die Terminalleitung für die Systemkonsole und andere Anmeldungen steuert. Vergewissern Sie sich, dass der Prozess `getty` auf allen Ausführungsebenen aktiviert ist. Weitere Informationen zur Datei `inittab` finden Sie im Abschnitt "inittab".
2. Verwenden Sie den Direktaufruf `telinit`, um auf das Menü **Systemausführungsebene festlegen** zuzugreifen. Weitere Informationen finden Sie in der Beschreibung des Befehls `telinit` in der Veröffentlichung "Commands Reference".
3. Geben Sie im Feld **Systemausführungsebene** die neue Systemausführungsebene ein.
4. Drücken Sie die Eingabetaste, um alle Einstellungen in dieser Prozedur zu implementieren.
Das System antwortet, indem es Ihnen mitteilt, welche Prozesse aufgrund der geänderten Ausführungsebene beendet oder gestartet werden, und indem es die folgende Nachricht anzeigt:

```
INIT: New run level: n
```

n steht für die Nummer der neuen Ausführungsebene.

Bootprozess

Es gibt drei Typen von Systembootprozessen und zwei Ressourcen, die erforderlich sind, um das Betriebssystem zu booten.

Während des Bootprozesses testet das System die Hardware, lädt und führt das Betriebssystem aus und konfiguriert Einheiten. Zum Booten des Betriebssystems sind die folgenden Ressourcen erforderlich:

- Ein *Boot-Image*, das geladen werden kann, nachdem die Maschine eingeschaltet oder zurückgesetzt wurde.
- Zugriff auf das Stammdateisystem (`/`) und auf das Dateisystem `/usr`.

Es gibt drei Typen von Systembootprozessen:

Eintrag	Beschreibung
Booten von der Festplatte	Eine Maschine wird für den normalen Betrieb gestartet.
Netz-IPL für Workstations ohne Plattenspeicher	Eine Workstation ohne Plattenspeicher oder eine Workstation mit systemreservierter Platte wird über Remote-Zugriff über ein Netz gestartet. Eine Maschine wird für den normalen Betrieb gestartet. Mindestens ein ferner Dateiserver stellt die Dateien und Programme bereit, die solche Workstations zum Booten benötigen.
Booten im Wartungsmodus	Eine Maschine wird im Wartungsmodus von einer Festplatte, über das Netz, von einem Band oder einer CD gestartet. In diesem Modus kann ein Systemadministrator Tasks ausführen, z. B. neue oder aktualisierte Software installieren oder Diagnoseprüfungen durchführen.

Beim Booten von einer Festplatte befindet sich das Boot-Image, das beim Installieren des Betriebssystems erstellt wurde, auf einer lokalen Festplatte. Während des Bootprozesses konfiguriert das System alle Einheiten, die in der Maschine enthalten sind, und initialisiert weitere Basissoftware, die für den Systembetrieb erforderlich ist (z. B. den Logical Volume Manager). Am Ende dieses Prozesses werden die Dateisysteme angehängt und können danach verwendet werden.

Dieselben allgemeinen Voraussetzungen gelten für Netzclients ohne Plattenspeicher. Auch sie setzen ein Boot-Image und Zugriff auf den Dateibaum des Betriebssystems voraus. Netzclients ohne Plattenspeicher haben keine lokalen Dateisysteme und erhalten alle Informationen über Remote-Zugriff.

Zugehörige Konzepte:

„Systemboot verarbeiten“ auf Seite 18

Die meisten Benutzer booten von der Festplatte, wenn Sie das System für den allgemeinen Betrieb star-

ten. Das System sucht alle erforderlichen Informationen für den Bootprozess auf seinem Plattenlaufwerk.

„Bootprozess im Wartungsmodus“ auf Seite 19

Gelegentlich kann es vorkommen, dass das System für die Ausführung spezieller Tasks, wie z. B. das Installieren neuer oder aktualisierter Software, die Durchführung von Diagnoseprüfungen, oder zu Wartungszwecken gebootet werden muss. In diesem Fall wird das System über ein bootfähiges Medium, wie z. B. eine CD-ROM, eine DVD, ein Bandlaufwerk, ein Netz oder ein Plattenlaufwerk, gestartet.

„RAM-Dateisystem“ auf Seite 20

Das RAM-Dateisystem gehört zum Boot-Image und ist vollständig speicherresident. Es enthält alle Programme, die die Fortsetzung des Bootprozesses ermöglichen. Die Dateien im RAM-Dateisystem richten sich nach dem Boottyp.

Systemboot verarbeiten:

Die meisten Benutzer booten von der Festplatte, wenn Sie das System für den allgemeinen Betrieb starten. Das System sucht alle erforderlichen Informationen für den Bootprozess auf seinem Plattenlaufwerk.

Wenn das System durch Einschalten des Netzschalters gestartet (Kaltstart) oder mit dem Befehl **reboot** oder **shutdown** erneut gestartet (Warmstart) wird, müssen eine Reihe von Ereignissen eintreten, bevor das System verwendet werden kann. Diese Ereignisse können in die folgenden Phasen eingeteilt werden:

Zugehörige Konzepte:

„Bootprozess“ auf Seite 17

Es gibt drei Typen von Systembootprozessen und zwei Ressourcen, die erforderlich sind, um das Betriebssystem zu booten.

Firmwarephase:

Die Firmware bereitet das System für das Laden und Ausführen des Betriebssystems vor.

Die Initialisierungsphase setzt sich aus den folgenden Schritten zusammen:

1. Die Firmware führt Basistests für die Systemressourcen durch, die für das Starten des Betriebssystems erforderlich sind.
2. Die Firmware überprüft die Bootliste des Benutzers, eine Liste mit den verfügbaren Booteinheiten. Diese Bootliste kann mit dem Befehl **bootlist** an die jeweiligen Anforderungen angepasst werden. Wenn die Bootliste des Benutzers im NVRAM (Non-Volatile Random Access Memory, nicht flüchtiger Arbeitsspeicher) nicht gültig ist oder wenn keine gültige Booteinheit gefunden wird, wird anschließend die Standardbootliste überprüft. In jedem Fall wird die erste gültige Booteinheit, die in der Bootliste gefunden wird, für den Systemstart verwendet. Wenn eine gültige Bootliste des Benutzers im NVRAM vorhanden ist, werden die Einheiten in der Liste nacheinander geprüft. Ist keine Bootliste des Benutzers vorhanden, werden alle Adapter und Einheiten im Bus geprüft. In jedem Fall werden die Einheiten fortlaufend geprüft, bis eine gültige Einheit Booteinheit für den Systemstart gefunden wird.

Anmerkung: Das System verwaltet eine Standardbootliste, die im NVRAM für einen normalen Bootvorgang gespeichert ist. Außerdem ist im NVRAM eine separate Bootliste für den Servicemodus gespeichert. Informationen darüber, wie Sie auf die Bootliste für den Servicemodus zugreifen können, finden Sie in den speziellen Hardwareanweisungen für Ihr Modell.

3. Wenn eine gültige Booteinheit gefunden wird, wird der erste Datensatz bzw. die erste Programmsektornummer (PSN, Program Sector Number) geprüft. Wenn der Bootsatz gültig ist, wird er in den Hauptspeicher eingelesen und dem IPL-Steuerblock im Hauptspeicher hinzugefügt. Zu den Daten im Hauptbootsatz gehören die Startadresse des Boot-Image auf der Booteinheit, die Größe des Boot-Image und Anweisungen, wohin das Boot-Image in den Hauptspeicher geladen werden soll.
4. Das Boot-Image wird ab der im NVRAM angegebenen Adresse sequenziell von der Booteinheit in den Hauptspeicher eingelesen. Das Platten-Boot-Image setzt sich aus dem Kernel, einem RAM-Dateisystem und angepassten Basiseinheitendaten zusammen.

5. Die Steuerung wird an den Kernel übergeben, woraufhin die Systeminitialisierung beginnt.
6. Der Kernel führt den Befehl **init** und damit Phase 1 des Scripts `rc.boot` aus.

Nach Abschluss der Kernelinitialisierung beginnt die Einheitenkonfiguration.

Phase für Konfiguration der Basiseinheiten:

Der Prozess **init** startet das Script `rc.boot`. In Phase 1 des Scripts `rc.boot` erfolgt die Konfiguration der Basiseinheiten.

Phase 1 des Scripts `rc.boot` umfasst die folgenden Schritte:

1. Das Boot-Script ruft das Programm **restbase** auf, damit aus den komprimierten angepassten Daten die angepasste ODM-Datenbank (Object Data Manager) im RAM-Dateisystem erstellt wird.
2. Das Boot-Script startet den Konfigurationsmanager, der auf die ODM-Konfigurationsregeln für Phase 1 zugreift, um die Basiseinheiten zu konfigurieren.
3. Der Konfigurationsmanager startet die Konfigurationsmethoden für **sys**, **bus**, **disk**, SCSI, den Logical Volume Manager (LVM) und die Datenträgergruppe `rootvg`.
4. Die Konfigurationsmethoden laden die Einheitentreiber, erstellen Gerätedateien und aktualisieren die angepassten Daten in der ODM-Datenbank.

System booten:

In der folgenden Prozedur ist die Systembootphase beschrieben.

1. Der Prozess **init** startet Phase 2 des Scripts `rc.boot`. Phase 2 des Scripts `rc.boot` umfasst die folgenden Schritte:
 - a. Programm **ipl_varyon** aufrufen, um die Datenträgergruppe `rootvg` zu aktivieren
 - b. Dateisysteme der Festplatte über ihre normalen Mountpunkte anhängen
 - c. Befehl **swapon** zum Starten des Paging ausführen
 - d. Angepasste Daten aus der ODM-Datenbank im RAM-Dateisystem in die ODM-Datenbank im Dateisystem der Festplatte kopieren
 - e. Script `rc.boot` beenden
2. Nach Abschluss von Phase 2 des Scripts `rc.boot` verlagert sich der Bootprozess vom RAM-Dateisystem auf die Dateisysteme, die auf der Festplatte gespeichert sind.
3. Der Prozess **init** führt anschließend die durch die Datensätze in der Datei `/etc/inittab` definierten Prozesse aus. Eine der Anweisungen in der Datei `/etc/inittab` leitet Phase 3 des Scripts `rc.boot` ein, die die folgenden Schritte umfasst:
 - a. Dateisystem `/tmp` der Festplatte anhängen
 - b. Phase 2 des Konfigurationsmanagers starten, um alle verbleibenden Einheiten zu konfigurieren
 - c. Mit dem Befehl **savebase** die angepassten Daten auf dem logischen Bootdatenträger speichern
 - d. Script `rc.boot` beenden

Am Ende dieses Prozesses ist das System aktiv und zur Verwendung bereit.

Bootprozess im Wartungsmodus:

Gelegentlich kann es vorkommen, dass das System für die Ausführung spezieller Tasks, wie z. B. das Installieren neuer oder aktualisierter Software, die Durchführung von Diagnoseprüfungen, oder zu Wartungszwecken gebootet werden muss. In diesem Fall wird das System über ein bootfähiges Medium, wie z. B. eine CD-ROM, eine DVD, ein Bandlaufwerk, ein Netz oder ein Plattenlaufwerk, gestartet.

Die Abfolge der Ereignisse beim Booten im Wartungsmodus ist nicht wesentlich anders als bei einem normalen Bootvorgang.

1. Die Firmware führt Basistests für die Systemressourcen durch, die für das Starten des Betriebssystems erforderlich sind.
2. Die Firmware überprüft die Bootliste des Benutzers. Mit dem Befehl **bootlist** können Sie die Bootliste an Ihre Anforderungen anpassen. Wenn die Bootliste des Benutzers im NVRAM (Non-Volatile Random Access Memory, nicht flüchtiger Arbeitsspeicher) nicht gültig ist oder wenn keine gültige Boot-einheit gefunden wird, wird anschließend die Standardbootliste überprüft. In jedem Fall wird die erste gültige Booteinheit, die in der Bootliste gefunden wird, zum Starten des Systems verwendet.

Anmerkung: Für normale Bootvorgänge verwaltet das Betriebssystem auch eine Standardbootliste und eine Benutzerbootliste im nicht flüchtigen Arbeitsspeicher (NVRAM, Non-Volatile Random Access Memory). Um das System im Wartungsmodus zu starten, werden ebenfalls separate Standardbootlisten und Benutzerbootlisten verwaltet.

3. Wenn eine gültige Booteinheit gefunden wird, wird der erste Datensatz bzw. die erste Programmsektornummer (PSN, Program Sector Number) geprüft. Wenn der Bootsatz gültig ist, wird er in den Hauptspeicher eingelesen und dem IPL-Steuerblock (Initial Program Load, einleitendes Programm-laden) im Hauptspeicher hinzugefügt. Zu den Daten im Hauptbootsatz gehören die Startadresse des Boot-Image auf der Booteinheit, die Größe des Boot-Image und der Abstand zum Einstiegspunkt, wenn sich das Boot-Image im Hauptspeicher befindet.
4. Das Boot-Image wird ab der im NVRAM angegebenen Adresse sequenziell von der Booteinheit in den Hauptspeicher eingelesen.
5. Die Steuerung wird an den Kernel übergeben, der mit der Ausführung der Programme im RAM-Dateisystem beginnt.
6. Der Inhalt der ODM-Datenbank bestimmt, welche Einheiten vorhanden sind, und der Befehl **cfgmgr** konfiguriert dynamisch alle gefundenen Einheiten, einschließlich aller Platten, auf denen das Stammdateisystem gespeichert werden soll.
7. Wenn eine CD-ROM, eine DVD, ein Band oder das Netz zum Booten des Systems verwendet wird, wird die Datenträgergruppe rootvg nicht aktiviert, weil rootvg möglicherweise nicht vorhanden ist (dies ist beispielsweise der Fall, wenn das Betriebssystem auf einem neuen System installiert wird). Zu diesem Zeitpunkt kann die Netzkonfiguration durchgeführt werden. Es findet kein Paging statt, wenn das System im Wartungsmodus gebootet wird.

Am Ende dieses Prozesses ist das System für Installations-, Wartungs- oder Diagnosearbeiten bereit.

Anmerkung: Wenn das System von der Festplatte gebootet wird, wird die Stammdatenträgergruppe angehängt. Dann werden Stammdateisystem und das Dateisystem /usr der Festplatte im Dateisystem des Arbeitsspeichers angehängt und es erscheint ein Menü, in dem Sie zwischen verschiedenen Diagnosemodi und dem Einzelbenutzermodus wählen können. Bei Auswahl des Einzelbenutzermodus können Sie den Bootprozess fortsetzen und im Einzelbenutzermodus arbeiten. In diesem Fall ist die Ausführungsebene von **init** auf "S" gesetzt. Anschließend ist das System für Wartungs- und Software-Updates sowie für die Ausführung des Befehls **bosboot** bereit.

Zugehörige Konzepte:

„Bootprozess“ auf Seite 17

Es gibt drei Typen von Systembootprozessen und zwei Ressourcen, die erforderlich sind, um das Betriebssystem zu booten.

RAM-Dateisystem:

Das RAM-Dateisystem gehört zum Boot-Image und ist vollständig speicherresident. Es enthält alle Programme, die die Fortsetzung des Bootprozesses ermöglichen. Die Dateien im RAM-Dateisystem richten sich nach dem Boottyp.

Ein RAM-Dateisystem, das für das Booten im Wartungsmodus verwendet wird, enthält möglicherweise nicht die Routinen für logische Datenträger, weil die Stammdatenträgergruppe rootvg unter Umständen nicht aktiviert werden muss. Beim Booten von einer Festplatte ist es jedoch wünschenswert, dass die

Stammdatenträgergruppe `rootvg` aktiviert ist und das Paging so bald wie möglich aktiviert wird. Obwohl es gewisse Unterschiede zwischen diesen beiden Bootszenarios gibt, weisen die Strukturen des RAM-Dateisystems keine großen Abweichungen auf.

Der Befehl `init`, der sich im Dateisystem des Arbeitsspeichers befindet, ist das während des Bootprozesses zu verwendende Interpreterprogramm für Basisbootbefehle. Dieses Interpreterprogramm für Bootbefehle steuert den Bootprozess durch Aufruf des Scripts `rc.boot`. Das Script `rc.boot` bestimmt, über welche Einheit die Maschine gestartet wurde. Die Booteinheit bestimmt, welche Einheiten im RAM-Dateisystem konfiguriert werden müssen. Wenn die Maschine über das Netz gestartet wird, müssen die Netzeinheiten konfiguriert werden, damit die Clientdateisysteme über Remote-Zugriff angehängt werden können. Wird die Maschine über ein Band, eine CD-ROM oder eine DVD gebootet, wird die Konsole für die Anzeige der Installationsmenüs für das Basisbetriebssystem konfiguriert. Nachdem das Script `rc.boot` die Booteinheit ermittelt hat, werden die entsprechenden Konfigurationsroutinen im RAM-Dateisystem aufgerufen. Das Script `rc.boot` wird vom Interpreterprogramm für Bootbefehle zweimal, d. h. in den beiden Konfigurationsphasen des Bootprogramms, aufgerufen. Wenn die Maschine über eine Platte oder das Netz gebootet wird, wird das Script `rc.boot` ein drittes Mal aufgerufen und zwar dann, wenn der Befehl `init` aufgerufen wird. Die Datei `inittab` enthält eine Zeilengruppe `rc.boot`, die die endgültige Konfiguration der Maschine durchführt.

Das RAM-Dateisystem jeder Booteinheit ist eindeutig, weil jeweils verschiedene Typen von Einheiten konfiguriert werden müssen. Jedem Typ von Booteinheit ist eine Prototypdatei zugeordnet. Die Prototypdatei ist eine Schablone von Dateien, aus denen sich das RAM-Dateisystem zusammensetzt. Der Befehl `bosboot` verwendet den Befehl `mkfs`, um das RAM-Dateisystem unter Verwendung der verschiedenen Prototypdateien zu erstellen. Ausführliche Informationen finden Sie in der Beschreibung des Befehls `bosboot`.

Zugehörige Konzepte:

„Bootprozess“ auf Seite 17

Es gibt drei Typen von Systembootprozessen und zwei Ressourcen, die erforderlich sind, um das Betriebssystem zu booten.

Fehlerbehebung beim Systemstart

Verwenden Sie die folgenden Methoden für die Behebung einiger der grundlegenden Probleme, die beim Starten Ihres Systems auftreten können. Wenn Ihr Problem in den verfügbaren Fehlerbehebungsinformationen nicht adressiert wird, wenden Sie sich an Ihren IBM Ansprechpartner.

Systeme, die nicht booten:

Wenn ein System nicht von der Festplatte bootet, können Sie trotzdem Zugriff auf das System erlangen, um das Problem zu bestimmen und zu beheben.

Wenn Sie ein System haben, das nicht von der Festplatte bootet, können Sie im Abschnitt "Troubleshooting your installation" in der Veröffentlichung *Installation and Migration* nachlesen, wie auf ein System zugegriffen wird, das nicht bootet.

Die in diesem Abschnitt beschriebene Prozedur gibt Ihnen die Möglichkeit, eine Eingabeaufforderung zu erhalten, so dass Sie versuchen können, die Daten vom System wiederherzustellen oder Korrekturmaßnahmen durchzuführen, damit das System wieder von der Festplatte booten kann.

Anmerkung:

- Diese Prozedur ist nur für erfahrene Systemmanager bestimmt, die wissen, wie ein System gebootet wird oder Daten von einem System wiederhergestellt werden, das nicht von der Festplatte bootet. Die meisten Benutzer sollten diese Prozedur nicht verwenden, sondern sich an ihren Kundendienst wenden.

- Diese Prozedur ist nicht für Systemmanager bestimmt, die soeben eine Neuinstallation durchgeführt haben, da in diesem Fall das System keine Daten enthält, die wiederhergestellt werden müssen. Wenn Sie das System nach einer Neuinstallation nicht von der Festplatte booten können, wenden Sie sich an Ihren Kundendienst.

Zugehörige Verweise:

„Diagnose von Bootproblemen“

Eine Vielzahl von Faktoren kann dazu führen, dass ein System nicht booten kann.

Diagnose von Bootproblemen:

Eine Vielzahl von Faktoren kann dazu führen, dass ein System nicht booten kann.

Einige dieser Faktoren sind im Folgenden aufgelistet:

- Hardwareprobleme
- Defekte Bootbänder oder -CDs
- Nicht ordnungsgemäß konfigurierte Netz-IPL-Server
- Beschädigte Dateisysteme
- Fehler in Scripts wie /sbin/rc.boot

Wenn der Bootprozess mit dem Referenzcode 2702 angehalten und die Nachricht "INSUFFICIENT ENTITLED MEMORY" angezeigt wird, verwenden Sie HMC (Hardware Management Console), und erhöhen Sie den zugesagten Hauptspeicher für diese Partition.

Zugehörige Konzepte:

„Systeme, die nicht booten“ auf Seite 21

Wenn ein System nicht von der Festplatte bootet, können Sie trotzdem Zugriff auf das System erlangen, um das Problem zu bestimmen und zu beheben.

Systemsicherung

Die nächste Überlegung nach der Inbetriebnahme des Systems muss der Sicherung der Dateisysteme, Verzeichnisse und Dateien gelten. Wenn Sie Ihre Dateisysteme sichern, können Sie Dateien oder Dateisysteme im Falle eines Festplattenfehlers wiederherstellen. Es gibt verschiedene Methoden für das Sichern von Informationen.

Das Sichern von Dateisystemen, Verzeichnissen und Dateien bedeutet eine erhebliche Investition, was Zeit- und Arbeitsaufwand angeht. Die Dateien auf dem Computer können jederzeit relativ leicht (versehentlich oder absichtlich) geändert oder gelöscht werden.

Achtung: Wenn durch einen Festplattenfehler die Daten auf der Platte zerstört werden, besteht die einzige Möglichkeit, die zerstörten Daten wiederherzustellen, darin, die Informationen von der Sicherungskopie zurückzuschreiben.

Durch eine systematische Vorgehensweise bei der Sicherung der Dateisysteme sollte es immer möglich sein, die aktuellste Version einer Datei oder eines Dateisystems ohne größere Schwierigkeit wiederherzustellen.

Es stehen verschiedene Methoden für das Sichern von Informationen zur Verfügung. Eine der am häufigsten verwendeten Methoden ist die *Sicherung nach Namen*, die *Dateinamensarchivierung* bzw. *reguläre Sicherung*. Es handelt sich hierbei um eine Kopie eines Dateisystems, eines Verzeichnisses oder einer Datei, die für die Dateiübertragung verwendet oder aufbewahrt wird, um für den Fall gerüstet zu sein, dass die Originaldaten versehentlich geändert oder gelöscht werden. Diese Sicherungsmethode wird angewendet, wenn das Flag **i** angegeben wird. Sie wird verwendet, um eine Sicherungskopie einzelner Dateien und Verzeichnisse zu erstellen. Diese Methode wird häufig von einzelnen Benutzern verwendet, die ihre Benutzerkonten sichern möchten.

Eine weitere, häufig verwendete Methode ist die *Sicherung nach I-Nodes*, die *Dateisystemarchivierung* bzw. *Archivierungssicherung*. Diese Sicherungsmethode wird angewendet, wenn das Flag **i** *nicht* angegeben wird. Sie wird verwendet, um eine Kopie der Daten für spätere Referenz, Protokollierungszwecke oder für die Wiederherstellung zu erstellen, falls die ursprünglichen Daten beschädigt werden oder verloren gehen. Bei dieser Sicherung wird eine Kopie eines vollständigen Dateisystems erstellt. Sie wird häufig von Systemadministratoren verwendet, um große Dateigruppen zu sichern, z. B. alle Benutzerkonten in /home. Eine Dateisystemsicherung ermöglicht problemlose inkrementelle Sicherungen. Bei einer inkrementellen Sicherung werden alle Dateien gesichert, die nach einer bestimmten früheren Sicherung geändert wurden.

Mit den Befehlen **compress** und **pack** können Sie Dateien für die Speicherung komprimieren, und mit den Befehlen **uncompress** und **unpack** können Sie wiederhergestellte Dateien entpacken. Das Packen und Entpacken von Dateien braucht zwar Zeit, aber nach dem Packen benötigen die Daten weniger Plattenspeicherplatz auf dem Sicherungsdatenträger. Weitere Informationen finden Sie in den Beschreibungen der Befehle **compress**, **pack**, **uncompress** und **unpack** in der Veröffentlichung "Commands Reference".

Es gibt mehrere Befehle, mit denen Sie Sicherungen und Archive erstellen können. Deshalb müssen die gesicherten Daten gekennzeichnet werden, damit erkennbar ist, welcher Befehl für die Sicherung verwendet und wie die Sicherung durchgeführt wurde (nach Namen oder nach Dateisystem).

Eintrag	Beschreibung
backup	Sichert Dateien nach Namen oder nach Dateisystem. Weitere Informationen finden Sie in der Beschreibung des Befehls backup in der Veröffentlichung "Commands Reference".
mksysb	Erstellt ein installierbares Image der Stammdatenträgergruppe (rootvg). Weitere Informationen finden Sie in der Beschreibung des Befehls mksysb in der Veröffentlichung "Commands Reference".
cpio	Kopiert Dateien in und aus Archivierungsspeichern. Weitere Informationen finden Sie im Abschnitt cpio in der Veröffentlichung "Commands Reference".
dd	Konvertiert und kopiert eine Datei. Dieser Befehl wird häufig verwendet, um Daten zu konvertieren und auf oder von Systemen mit anderen Betriebssystemen zu kopieren, z. B. Großrechner. Der Befehl dd gruppiert mehrere Dateien nicht zu einem Archiv. Er wird verwendet, um Daten zu bearbeiten und zu verschieben. Weitere Informationen finden Sie in der Beschreibung des Befehls dd in der Veröffentlichung "Commands Reference".
tar	Erstellt oder bearbeitet Archive mit tar-Format. Weitere Informationen finden Sie in der Beschreibung des Befehls tar in der Veröffentlichung "Commands Reference".
rdump	Sichert Dateien nach Dateisystem auf einer Einheit eines fernen Systems. Weitere Informationen finden Sie in der Beschreibung des Befehls rdump in der Veröffentlichung "Commands Reference".
pax	(POSIX-konformes Archivierungsdienstprogramm) Liest und schreibt Archive mit tar - und cpio -Format. Weitere Informationen finden Sie in der Beschreibung des Befehls pax in der Veröffentlichung "Commands Reference".

Zugehörige Konzepte:

„Sicherung für BSD-4.3-Systemmanager“ auf Seite 348
 BSD-4.3-Systemmanager können Daten sichern.

Zugehörige Tasks:

„Benutzerdateien und -dateisysteme sichern“ auf Seite 28
 Für das Sichern von Dateien und Dateisystemen stehen zwei Prozeduren zur Verfügung: die SMIT-Direktaufrufe **smit backfile** und **smit backfilesys** sowie der Befehl **backup**.

Sicherungskonzepte

Bevor Sie mit dem Sichern Ihrer Daten beginnen, müssen Sie sich mit den Typen von Daten, Richtlinien und Medien vertraut machen, die Sie verwenden können.

Sicherungsrichtlinien:

Es gibt keine Sicherungsrichtlinie, die den Anforderungen aller Benutzer gleichermaßen entspricht. Wenn eine Richtlinie für ein System mit einem Benutzer ideal ist, muss dies nicht gleichermaßen für ein System gelten, das hundert Benutzer bedient. Ebenso ist eine Richtlinie für ein System, auf dem täglich viele Dateien geändert werden, nicht angebracht für ein System, auf dem die Daten selten geändert werden.

Welche Sicherungsstrategie für Ihren Standort auch immer die geeignete ist, es ist wichtig, dass eine vorhanden ist und dass die Sicherungen häufig und regelmäßig vorgenommen werden. Ein Datenverlust lässt sich nur schwer verkraften, wenn keine taugliche Sicherungsstrategie implementiert ist.

Nur Sie können entscheiden, welche Sicherungsrichtlinie für Ihr System am besten ist. Die folgenden Richtlinien können jedoch als Unterstützung dienen:

- **Stellen Sie sicher, dass Sie auch größere Datenverluste wiederherstellen können.**

Kann Ihr System nach einem einzigen Festplattenfehler weiter betrieben werden? Können Sie Ihr System wiederherstellen, wenn alle Festplatten defekt sind? Könnten Sie Ihr System wiederherstellen, falls Ihre Sicherungsdisketten oder Ihr Sicherungsband durch Feuer oder Diebstahl verloren gehen? Wenn die Daten verloren gegangen sind, wie schwierig wäre es, sie wiederherzustellen? Durchdenken Sie alle möglichen, auch die unwahrscheinlichen Fälle für einen Datenverlust, und legen Sie eine Sicherungsrichtlinie fest, die es Ihnen in jedem Fall ermöglicht, Ihr System wiederherzustellen.

- **Überprüfen Sie Ihre Sicherungen in regelmäßigen Abständen.**

Sicherungsdatenträger und die zugehörige Hardware können unzuverlässig sein. Eine umfangreiche Bibliothek mit Sicherungsbändern oder -disketten ist nutzlos, wenn die Daten nicht gelesen und auf eine Festplatte zurückgeschrieben werden können. Um sicherzustellen, dass Ihre Sicherungen verwendbar sind, zeigen Sie in regelmäßigen Abständen das Inhaltsverzeichnis auf dem Sicherungsband an (mit **restore -T** oder **tar -t** für Archivierungsbänder). Wenn Sie Disketten für Ihre Sicherungen verwenden und mehrere Diskettenlaufwerke haben, lesen Sie die Disketten von einem anderen Laufwerk als dem Laufwerk, das Sie für die Erstellung der Sicherung verwendet haben. Als zusätzliche Sicherheit kann jede Sicherung der Stufe 0 mit einem zweiten Datenträgersatz wiederholt werden. Wenn Sie eine Datenstrombandeinheit verwenden, können Sie mit dem Befehl **tapechk** grundlegende Konsistenzprüfungen für das Band durchführen. Weitere Informationen finden Sie in den Beschreibungen der Befehle **restore -T**, **tar -t** und **tapechk** in der Veröffentlichung "Commands Reference".

- **Bewahren Sie alte Sicherungen auf.**

Die Sicherungsdatenträger sollten in regelmäßigen Abständen wiederverwendet werden. Es sollten jedoch nicht alle Sicherungsdatenträger wiederverwendet werden. Manchmal kann es Monate dauern, bis Sie oder ein anderer Benutzer des Systems feststellt, dass eine wichtige Datei beschädigt ist oder fehlt. Bewahren Sie für solche Fälle alte Sicherungen auf. Das folgende Beispiel zeigt eine Möglichkeit der turnusmäßigen Verwendung von Sicherungsbändern und -disketten:

- Einmal pro Woche alle Disketten mit Tagessicherungen mit Ausnahme der vom Freitag wiederverwenden.
- Einmal pro Monat alle Disketten mit den Sicherungen von Freitagen mit Ausnahme der vom letzten Freitag des Monats wiederverwenden. Dadurch stehen jederzeit die Sicherungen der vorangegangenen Freitage zur Verfügung.
- Einmal im Quartal alle Monatsdisketten mit Ausnahme der letzten wiederverwenden. Heben Sie die Diskette mit der letzten Monatssicherung jedes Quartals grundsätzlich auf, vorzugsweise in einem anderen Gebäude.

- **Überprüfen Sie vor der Sicherung die Dateisysteme.**

Die Sicherung eines beschädigten Dateisystems kann wertlos sein. Deshalb empfiehlt es sich, vor dem Erstellen der Sicherungen die Integrität des Dateisystems mit dem Befehl **fsck** zu überprüfen. Weitere Informationen finden Sie in der Beschreibung des Befehls **fsck** in der Veröffentlichung "Commands Reference".

- **Vergewissern Sie sich, dass die Dateien während der Durchführung der Sicherung nicht im Gebrauch sind.**

Verwenden Sie das System während der Durchführung der Sicherung nicht. Wenn das System im Gebrauch ist, können sich Dateien während der Sicherung ändern, und in diesem Fall ist die Sicherungskopie nicht präzise.

- **Sichern Sie Ihr System, bevor größere Änderungen am System vorgenommen werden.**

Es empfiehlt sich grundsätzlich, das gesamte System zu sichern, bevor Hardwaretests oder Reparaturarbeiten durchgeführt oder neue Einheiten, Programme oder sonstige Zusatzeinrichtungen des Systems installiert werden.

- **Weitere Faktoren**

Berücksichtigen Sie bei der Planung und Implementierung Ihrer Sicherheitsstrategie die folgenden Faktoren:

- Wie oft werden die Daten geändert? Die Daten des Betriebssystems ändern sich nicht sehr häufig und müssen deshalb auch nicht sehr häufig gesichert werden. Benutzerdaten hingegen werden häufig geändert und sollten daher häufig gesichert werden.
- Wie viele Benutzer arbeiten am System? Nach der Anzahl der Benutzer richtet sich die benötigte Anzahl von Speichermedien und die Häufigkeit der Sicherungen.
- Wie schwer wäre es, die Daten wiederherzustellen? Es ist zu beachten, dass manche Daten ohne Sicherung nicht wiederhergestellt werden können.

Es ist extrem wichtig, eine Sicherungsstrategie zu haben, um die Verfügbarkeit Ihrer Daten zu gewährleisten. Durch Auswertung Ihrer Standortanforderungen können Sie die Sicherungsrichtlinien ermitteln, die sich am besten für Sie eignen. Erstellen Sie häufig und regelmäßig Sicherungen der Benutzerdaten. Die Wiederherstellung nach Datenverlusten gestaltet sich schwierig, wenn keine optimale Sicherungsstrategie implementiert ist.

Anmerkung: Für die Sicherung benannter Pipes (FIFO-Sonderdateien) können die Pipes geschlossen oder geöffnet sein. Die Wiederherstellung schlägt jedoch fehl, wenn geöffnete benannte Pipes gesichert werden. Wenn Sie eine FIFO-Sonderdatei wiederherstellen, ist der I-Node der Datei alles, was benötigt wird, um die Datei erneut zu erstellen, weil dieser alle Kenndaten der Datei enthält. Der Inhalt der benannten Pipe ist für die Wiederherstellung nicht relevant. Deshalb wird während der Sicherung die Dateigröße null (alle FIFOs geschlossen) und erst nach Abschluss der Sicherung die tatsächliche Dateigröße angezeigt.

Achtung: Die Systemsicherungs- und Systemwiederherstellungsprozeduren setzen voraus, dass das System auf demselben Plattformtyp wiederhergestellt wird, auf dem die Sicherung erstellt wurde. Insbesondere die CPU und die E/A-Systemplatinen müssen denselben Typ haben.

Sicherungsdatenträger:

Sie können verschiedene Arten von Sicherungsdatenträgern verwenden. Die für die jeweilige Systemkonfiguration zur Verfügung stehenden Arten von Sicherungsdatenträgern hängen sowohl von der Software als auch von der Hardware ab.

Sie können verschiedene Typen von Sicherungsdatenträgern verwenden. Die für die jeweilige Systemkonfiguration zur Verfügung stehenden Typen von Sicherungsdatenträgern richten sich sowohl nach der Software als auch nach der Hardware. Am häufigsten verwendet werden Bänder (8-mm-Bänder und Bänder mit 9 Spuren), Disketten (5,25-Zoll- und 3,5-Zoll-Disketten), ferne Archive und alternative lokale Festplatten. Sofern Sie mit dem Befehl **backup -f** keine andere Einheit angeben, schreibt der Befehl **backup** seine Ausgabe automatisch nach `/dev/rfd0`, d. h. das Diskettenlaufwerk.

Achtung: Bei der Ausführung des Befehls **backup** werden alle bereits auf dem ausgewählten Sicherungsdatenträger gespeicherten Daten gelöscht.

Disketten

Standardmäßig werden Disketten als Sicherungsdatenträger verwendet. Sofern Sie mit dem Befehl **backup -f** keine andere Einheit angeben, schreibt der Befehl **backup** seine Ausgabe automatisch auf die Einheit `/dev/rfd0`, d. h. das Diskettenlaufwerk. Wenn Sie Daten auf der Standardbandeinheit sichern möchten, geben Sie `/dev/rmt0` ein, und drücken Sie anschließend die Eingabetaste.

Gehen Sie sorgfältig mit den Disketten um. Jede Informationseinheit belegt einen so kleinen Bereich auf der Diskette, dass selbst kleine Kratzer, Staub, Lebensmittelflecken oder Tabakkrümel die Informationen wertlos machen können. Beachten Sie folgende Richtlinien:

- Die Aufzeichnungsflächen nicht berühren.
- Disketten nicht in die Nähe von Magneten oder Geräten mit magnetischen Feldern, wie z. B. Telefonen, Diktiergeräten oder elektronischen Taschenrechnern, legen.
- Disketten keinen extremen Temperaturen aussetzen. Der empfohlene Temperaturbereich liegt zwischen 10 und 60 Grad Celsius.
- Eine sorgfältige Behandlung der Disketten schützt vor Datenverlusten.
- In regelmäßigen Abständen Sicherungskopien der Disketten erstellen.

Achtung: Zur erfolgreichen Sicherung der Daten muss im verwendeten Laufwerk der richtige Diskettentyp eingelegt werden. Wenn Sie eine falsche Diskette in das 3,5-Zoll-Laufwerk einlegen, können Sie damit die auf der Diskette gespeicherten Daten zerstören.

Die folgenden 3,5-Zoll-Disketten können im Laufwerk eingelegt werden:

- Speicherkapazität von 1 MB (für ca. 720 KB Daten)
- Speicherkapazität von 2 MB (für ca. 1,44 MB Daten)

Bänder

Aufgrund ihrer hohen Speicherkapazität und Haltbarkeit werden Bänder häufig zum Speichern großer oder vieler Dateien verwendet, z. B. für Archivkopien von Dateisystemen. Bänder eignen sich auch, wenn viele Dateien von einem System auf ein anderes übertragen werden müssen. Bänder werden im Allgemeinen nicht für die Sicherung von Dateien verwendet, auf die häufig zugegriffen wird. In diesem Fall bieten andere Speichermedien erheblich kürzere Zugriffszeiten.

Banddateien werden mit den Befehlen **backup**, **cpio** und **tar** erstellt, die ein Bandlaufwerk öffnen, darauf schreiben und es anschließend wieder schließen.

Sicherungsstrategie:

Es gibt zwei Methoden für die Sicherung großer Datenmengen.

- Vollständige Systemsicherung
- Inkrementelle Sicherung

Für das Verständnis dieser beiden Sicherungstypen und für die Entscheidung darüber, welcher der richtige für einen Standort oder ein System ist, ist es wichtig, die Dateisystemstruktur und die Datenplatzierung zu verstehen. Nachdem Sie sich für eine Strategie für die Datenplatzierung entschieden haben, können Sie eine Sicherungsstrategie für diese Daten entwickeln.

Zugehörige Tasks:

„Geplante Sicherungen implementieren“ auf Seite 47

Diese Prozedur beschreibt, wie Sie ein Script entwickeln und verwenden, um eine wöchentliche vollständige Sicherung und tägliche inkrementelle Sicherungen von Benutzerdateien durchzuführen.

Systemdaten und Benutzerdaten:

Daten sind Programme oder Text und werden für diese Beschreibung in zwei Klassen unterteilt:

- Systemdaten: umfassen das Betriebssystem und dessen Erweiterungen. Diese Daten müssen immer in Systemdateisystemen, d. h. / (Stammdateisystem), /usr, /tmp, /var usw., gespeichert werden.
- Benutzerdaten sind gewöhnlich lokale Dateien, die Einzelpersonen benötigen, um ihre jeweiligen Tasks auszuführen. Diese Daten müssen im Dateisystem /home oder in Dateisystemen gespeichert werden, die speziell für Benutzerdaten erstellt werden.

Benutzerprogramme und Text dürfen nicht in Dateisystemen gespeichert werden, die für Systemdaten bestimmt sind. Beispielsweise kann ein Systemmanager ein neues Dateisystem erstellen und über /local anhängen. Eine Ausnahme ist das Dateisystem /tmp, das als temporärer Speicher für System- und Benutzerdaten verwendet wird.

Sicherungen:

Im Allgemeinen werden Sicherungen von Benutzer- und Systemdaten aufbewahrt für den Fall, dass Daten versehentlich entfernt werden oder dass eine Platte defekt ist. Sicherungen lassen sich einfacher verwalten, wenn Benutzerdaten von Systemdaten getrennt aufbewahrt werden.

Aus den folgenden Gründen sollten Systemdaten von Benutzerdaten getrennt aufbewahrt werden:

- Benutzerdaten ändern sich in der Regel häufiger als Betriebssystemdaten. Sicherungsbilder sind viel kleiner, wenn die Systemdaten nicht in demselben Image wie die Benutzerdaten gesichert werden. Nach der Anzahl der Benutzer richtet sich die benötigte Anzahl von Speichermedien und die Häufigkeit der Sicherungen.
- Benutzerdaten lassen sich schneller und einfacher wiederherstellen, wenn sie gesondert aufbewahrt werden. Wenn das Betriebssystem zusammen mit den Benutzerdaten wiederhergestellt wird, bedeutet dies zusätzlichen Zeit- und Arbeitsaufwand. Dies liegt daran, dass die für die Wiederherstellung der Betriebssystemdaten verwendete Methode das System von einem austauschbaren Datenträger (Band oder CD) startet und die Systemsicherung installiert.

Wenn Sie die Systemdaten sichern möchten, müssen Sie mit dem Befehl **umount** alle Benutzerdateisysteme, einschließlich des Dateisystems /home, abhängen. Wenn diese Dateisysteme im Gebrauch sind, können Sie sie nicht abhängen. Planen Sie die Sicherungen deshalb für Zeiten geringer Systemnutzung, damit die Dateisysteme abgehängt werden können. Wenn die Dateisysteme mit Benutzerdaten angehängt bleiben, werden sie zusammen mit den Betriebssystemdaten gesichert. Verwenden Sie den Befehl **mount**, um sicherzustellen, dass nur die Dateisysteme mit dem Betriebssystem angehängt sind.

Die einzigen angehängten Dateisysteme sind /, /usr, /var und /tmp, und das Ergebnis des Befehls **mount** kann der folgenden Ausgabe gleichen:

node	mounted	mounted over	vfs	date	options
/dev/hd4	/		jfs	Jun 11 10:36	rw,log=/dev/hd8
/dev/hd2	/usr		jfs	Jun 11 10:36	rw,log=/dev/hd8
/dev/hd9var	/var		jfs	Jun 11 10:36	rw,log=/dev/hd8
/dev/hd	/tmp		jfs	Jun 11 10:36	rw,log=/dev/hd8

Wenn Sie sicher sind, dass alle Dateisysteme abgehängt sind, können Sie die Betriebssystemdaten sichern.

Nachdem Sie die Sicherung des Betriebssystems abgeschlossen haben, hängen Sie das Benutzerdateisystem mit dem Befehl **smit mount** an. Anschließend können Sie je nach Bedarf Dateien, Dateisysteme oder weitere Datenträgergruppen sichern.

Zugehörige Konzepte:

„Sicherung von Systemimages und benutzerdefinierten Datenträgergruppen“ auf Seite 43

Die Stammdatenträgergruppe (rootvg) ist auf einer Festplatte oder einer Gruppe von Festplatten gespeichert und enthält Startdateien, das Basisbetriebssystem (BOS, Base Operating System), Konfigurationsdaten und alle optionalen Softwareprodukte. Eine *benutzerdefinierte Datenträgergruppe* (auch *Nicht-rootvg-Datenträgergruppe* genannt) enthält normalerweise Datendateien und Anwendungssoftware.

Systemreplikation (Klonen):

Beim Klonen werden Konfigurationsdaten zusammen mit Benutzer- oder Systemdaten gespeichert. Sie können beispielsweise ein System oder eine Datenträgergruppe replizieren. Dies wird zuweilen als Klonen bezeichnet.

Anschließend können Sie dieses Image auf einem anderen System installieren und wie das erste System verwenden. Der Befehl **mksysb** wird zum Klonen der Datenträgergruppe **rootvg** verwendet, die das Betriebssystem enthält. Mit dem Befehl **savevg** kann eine normale Datenträgergruppe geklont werden.

Befehlsübersicht für Sicherungsdateien und Speichermedien:

Es sind Befehle für das Sichern von Dateien und Speichern von Daten verfügbar.

Eintrag	Beschreibung
backup	Sichert Dateien und Dateisysteme.
compress	Komprimiert und dekomprimiert Daten.
cpio	Kopiert Dateien in und aus Archivierungsspeicher(n) und Verzeichnisse(n).
fdformat	Formatiert Disketten.
flcopy	Kopiert Daten auf und von Disketten.
format	Formatiert Disketten.
fsck	Überprüft Dateisysteme auf Konsistenz und repariert Dateisysteme interaktiv.
pack	Komprimiert Dateien.
restore	Kopiert zuvor gesicherte Dateisysteme oder Dateien, die mit dem Befehl backup erstellt wurden, von einer lokalen Einheit.
tapechk	Überprüft die Konsistenz von Datenstrombandeinheiten.
tar	Bearbeitet Archive.
tcopy	Kopiert Magnetbänder.
uncompress	Komprimiert und dekomprimiert Daten.
unpack	Dekomprimiert Dateien.

System Sicherungen verwalten

Es sind mehrere Methoden für die Sicherung Ihres Systems und die Wiederherstellung einer Systemsicherung verfügbar.

Benutzerdateien und -dateisysteme sichern:

Für das Sichern von Dateien und Dateisystemen stehen zwei Prozeduren zur Verfügung: die SMIT-Direktaufrufe **smit backfile** und **smit backfilesys** sowie der Befehl **backup**.

- Wenn Sie Dateisysteme nach I-Node sichern möchten und diese Dateisysteme im Gebrauch sind, hängen Sie sie zuerst ab, um Inkonsistenzen zu vermeiden.
Achtung: Wenn Sie versuchen, ein angehängtes Dateisystem zu sichern, wird eine Warnung angezeigt. Der Befehl **backup** wird zwar fortgesetzt, aber es können Inkonsistenzen im Dateisystem auftreten. Diese Warnung gilt nicht für das Stammdateisystem (/).
- Stellen Sie sicher, dass seit der letzten Reinigung der Sicherungseinheit nicht zu viel Zeit vergangen ist, um Fehler zu verhindern.

Zum Sichern von Benutzerdateien und -dateisystemen können Sie die SMIT-Direktaufrufe **smit backfile** und **smit backfilesys** verwenden.

Mit der Schnittstelle SMIT können Sie einzelne und kleine Dateisysteme nach Namen sichern, z. B. /home auf Ihrem lokalen System. Beachten Sie, dass SMIT Archivierungen nur in dem Format vornehmen kann, das vom Befehl **backup** unterstützt wird. Außerdem sind nicht alle Flags des Befehls **backup** in SMIT verfügbar. SMIT kann blockieren, wenn mehrere Bänder oder Platten für die Sicherung erforderlich sind. Weitere Informationen hierzu finden Sie in der Beschreibung des Befehls **backup** in der Veröffentlichung *Commands Reference, Volume 1*.

Verwenden Sie den Befehl **backup**, wenn Sie große oder mehrere Dateisysteme sichern möchten. Sie können eine Stufennummer angeben, um zu steuern, wie viel Daten gesichert werden. Die Nummer 0 steht für eine vollständige Sicherung, die Stufen 1-9 für eine inkrementelle Sicherung. Stufennummern für Sicherungen können nur mit dem Befehl **backup** angegeben werden.

Der Befehl **backup** erstellt Kopien in einem der beiden folgenden Sicherungsformate:

- Wenn das Flag **-i** angegeben wird, werden bestimmte Dateien nach Namen gesichert.
- Wenn die Parameter **-Stufe** und **Dateisystem** angegeben werden, werden vollständige Dateisysteme nach I-Node gesichert. Das Dateisystem wird defragmentiert, wenn es aus der Sicherung wiederhergestellt wird.

Achtung: Die Sicherung nach I-Node funktioniert nicht ordnungsgemäß für Dateien, die eine Benutzer-ID (UID) oder eine Gruppen-ID (GID) größer als 65535 haben. Diese Dateien werden mit abgeschnittener UID oder GID gespeichert und haben deshalb nach der Wiederherstellung ungültige UID- bzw. GID-Attribute. In diesen Fällen müssen Sie eine Sicherung nach Namen durchführen.

Tasks für das Sichern von Benutzerdateien oder -dateisystemen

Task	SMIT-Direktaufruf	Befehl oder Datei
Benutzerdateien sichern	smit backfile	<ol style="list-style-type: none"> 1. Melden Sie sich an Ihrem Benutzeraccount an. 2. Führen Sie die Sicherung durch: <code>find . -print backup -ivf /dev/rmt0</code>
Benutzerdateisysteme sichern	smit backfilesys	<ol style="list-style-type: none"> 1. Hängen Sie die Dateisysteme ab, die Sie sichern möchten. Geben Sie beispielsweise <code>umount all</code> oder <code>umount /home /filesys1</code> ein. 2. Überprüfen Sie die Dateisysteme, z. B. <code>fsck /home /filesys1</code>. 3. Führen Sie die Sicherung nach I-Node durch, z. B. <code>backup -5 -uf/dev/rmt0 /home/libr</code>. 4. Stellen Sie die Dateien mit dem folgenden Befehl wieder her: restore -t.

Anmerkung: Wird von diesem Befehl eine Fehlermeldung generiert, muss die gesamte Sicherung wiederholt werden.

Zugehörige Konzepte:

„Systemsicherung“ auf Seite 22

Die nächste Überlegung nach der Inbetriebnahme des Systems muss der Sicherung der Dateisysteme, Verzeichnisse und Dateien gelten. Wenn Sie Ihre Dateisysteme sichern, können Sie Dateien oder Dateisysteme im Falle eines Festplattenfehlers wiederherstellen. Es gibt verschiedene Methoden für das Sichern von Informationen.

Gesicherte Dateien wiederherstellen:

Nachdem die Daten ordnungsgemäß gesichert wurden, gibt es verschiedene Methoden, mit denen Sie die Daten wiederherstellen können. Die zu verwendende Methode richtet sich nach dem Typ des verwendeten Sicherungsbefehls.

Für eine ordnungsgemäße Wiederherstellung müssen Sie wissen, wie die Sicherung bzw. das Archiv erstellt wurde. Jede Sicherungsprozedur liefert Informationen zur Wiederherstellung der Daten. Wenn Sie beispielsweise den Befehl **backup** verwenden, können Sie eine Sicherung nach Dateisystem oder nach Namen angeben. Diese Sicherung muss so wiederhergestellt werden, wie sie erstellt wurde, d. h. entweder nach Dateisystem oder nach Namen. Weitere Informationen finden Sie in der Beschreibung des Befehls **backup** in der Veröffentlichung "Commands Reference".

Es stehen verschiedene Befehle zur Verfügung, mit denen gesicherte Daten wiederhergestellt werden können, wie z. B.:

Eintrag	Beschreibung
restore	Kopiert Dateien, die mit dem Befehl backup erstellt wurden. Weitere Informationen zur Verwendung dieses Befehls finden Sie im folgenden Abschnitt.
rrestore	Kopiert Dateisysteme, die auf einem fernen System gesichert wurden, auf das lokale System. Weitere Informationen finden Sie im Abschnitt rrestore in der Veröffentlichung "Commands Reference".
cpio	Kopiert Dateien in und aus Archivierungsspeichern. Weitere Informationen finden Sie im Abschnitt cpio in der Veröffentlichung "Commands Reference".
tar	Erstellt oder bearbeitet tar -Archive. Weitere Informationen finden Sie in der Beschreibung des Befehls tar in der Veröffentlichung "Commands Reference".
pax	(POSIX-konformes Archivierungsdienstprogramm) Liest und schreibt Archive mit tar - und cpio -Format. Weitere Informationen finden Sie in der Beschreibung des Befehls pax in der Veröffentlichung "Commands Reference".

In den folgenden Abschnitten werden die Befehle **restore** und **smit** beschrieben.

Anmerkung:

- Für die Wiederherstellung der Dateien muss dieselbe Methode verwendet werden, die auch für die Sicherung verwendet wurde. Wenn ein Dateisystem beispielsweise nach Namen gesichert wurde, muss es auch nach Name wieder zurückgeschrieben werden.
- Sollten mehrere Disketten benötigt werden, liest der Befehl **restore** die angehängte Diskette, fordert Sie dann zum Einlegen der nächsten Diskette auf und wartet auf Ihre Antwort. Drücken Sie nach dem Einlegen der neuen Diskette die Eingabetaste, um mit dem Zurückschreiben der Datei fortzufahren.

Dateien mit dem Befehl **restore** wiederherstellen

Mit dem Befehl **restore** können Sie Dateien, die mit dem Befehl **backup** geschrieben wurden, lesen und auf Ihrem lokalen System wiederherstellen.

Beispiele:

- Geben Sie zum Auflisten der zuvor gesicherten Dateien den folgenden Befehl ein:
`restore -T`

Die Informationen werden von der Standardsicherungseinheit `/dev/rfd0` gelesen. Wurden einzelne Dateien gesichert, werden nur die Dateinamen angezeigt. Wurde ein gesamtes Dateisystem gesichert, wird zusätzlich die I-Node-Nummer angezeigt.

- Geben Sie zum Wiederherstellen der Dateien im Hauptdateisystem den folgenden Befehl ein:
`restore -x -v`

Das Flag `-x` extrahiert alle Dateien vom Sicherungsdatenträger und stellt sie an den entsprechenden Positionen im Dateisystem wieder her. Mit dem Flag `-v` wird nach jeder zurückgeschriebenen Datei ein Fortschrittsbericht angezeigt. Beim Wiederherstellen einer Dateisystemsicherung werden die Dateien zusammen mit den entsprechenden I-Node-Nummern angegeben. Sonst werden nur die Namen angezeigt.

- Geben Sie beispielsweise Folgendes ein, wenn Sie die Datei `/home/mike/manual/chap1` kopieren möchten:
`restore -xv /home/mike/manual/chap1`

Dieser Befehl extrahiert die Datei `/home/mike/manual/chap1` vom Sicherungsdatenträger und schreibt sie zurück. Die Datei `/home/mike/manual/chap1` muss einen Namen haben, den der Befehl **restore -T** anzeigen kann.

- Geben Sie beispielsweise Folgendes ein, um alle Dateien aus einem Verzeichnis mit dem Namen `manual` zu kopieren:
`restore -xdv manual`

Mit diesem Befehl werden das Verzeichnis `manual` und die darin enthaltenen Dateien zurückgeschrieben. Sollte das Verzeichnis nicht vorhanden sein, wird ein Verzeichnis `manual` im aktuellen Verzeichnis erstellt, in das dann die wiederhergestellten Dateien geschrieben werden.

Die vollständige Syntax des Befehls **restore** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

Dateien mit dem Befehl **smit** wiederherstellen

Mit dem Befehl **smit** können Sie den Befehl **restore** ausführen, der Dateien, die mit dem Befehl **backup** geschrieben wurden, liest und auf Ihrem lokalen System wiederherstellt.

1. Geben Sie an der Eingabeaufforderung Folgendes ein:
`smit restore`
2. Geben Sie im Feld **Zielverzeichnis** den gewünschten Wert ein. In dieses Verzeichnis werden die Dateien zurückgeschrieben.
3. Geben Sie im Feld **Sicherungseinheit** oder im Feld **Datei** den Namen der Ausgabeinheit, wie z. B. den Namen einer Magnetbandeinheit, ein.

`/dev/rmt0`

Ist die Einheit nicht verfügbar, wird eine Nachricht wie die folgende angezeigt:

`/dev/rmtX kann nicht geöffnet
werden. Datei oder Verzeichnis ist nicht vorhanden.`

Diese Nachricht gibt an, dass das System den Einheitentreiber nicht erreichen kann, weil keine Datei für **rmtX** im Verzeichnis `/dev` vorhanden ist. Im Verzeichnis `/dev` sind nur Elemente im Status `verfügbar` enthalten.

4. Es wird empfohlen, für das Feld **Anzahl der in einer Eingabeoperation zu lesenden Blöcke** den Standardwert zu verwenden.
5. Drücken Sie die Eingabetaste, um das angegebene Dateisystem oder Verzeichnis zurückzuschreiben.

Ein fernes Archiv erstellen:

Verwenden Sie diese Prozedur, um Dateien auf einer fernen Bandeinheit zu archivieren.

Aktive AIX-Systeme können eine ferne Bandeinheit nicht so anhängen, als wäre sie eine lokale Einheit des Systems. Mit dem Befehl **rsh** können Daten jedoch an die Bandeinheit einer fernen Maschine gesendet werden. Die folgende Prozedur schreibt nur auf ein Band. Für Archive mit mehreren Bändern ist eine spezielle Anwendungssoftware erforderlich.

Für die folgende Prozedur werden folgende Informationen benötigt:

Blockgröße

Gibt die Blockgröße für die Zielbandeinheit an.

ferner_Host

Der Name des Zielsystems (das System mit der Bandeinheit).

Quellenhost

Der Name des Quellensystems (das zu archivierende System).

/dev/rmt0

Der Name der fernen Bandeinheit.

Pfadname

Der vollständige Pfadname eines erforderlichen Verzeichnisses oder einer erforderlichen Datei.

In den folgenden Anweisungen wird Root als lokaler und als ferner Benutzer angenommen.

1. Vergewissern Sie sich, dass Sie Zugriff auf das ferne System haben. Das Quellsystem muss Zugriff auf das System mit dem Bandlaufwerk haben. (Auf das Zielsystem kann jeder der definierten Benutzer auf diesem System zugreifen, aber der Benutzername muss für viele der folgenden Schritte Rootberechtigung besitzen.)
2. Erstellen Sie in dem von Ihnen bevorzugten Editor eine Datei mit dem Namen `.rhosts` im Stammverzeichnis (`/`) des Zielsystems, die es dem Quellsystem ermöglicht, auf das Zielsystem zuzugreifen. Sie müssen dieser Datei den berechtigten Hostnamen und die Benutzer-ID hinzufügen. Mit dem folgenden Befehl können Sie den Namen des Quellsystems für die Datei `.rhosts` feststellen:
`host IP-Adresse_des_Quellsystems`

Für dieses Beispiel fügen Sie der Datei `.rhosts` die folgende Zeile hinzu:

```
Quellenhost.mynet.com root
```

3. Speichern Sie die Datei, und ändern Sie anschließend die Dateiberechtigungen mit dem folgenden Befehl:
`chmod 600 .rhosts`
4. Verwenden Sie den Befehl `rsh`, um Ihren Zugriff vom Quellsystem aus zu testen. Beispiel:
`rsh remotehost`

Wenn alles richtig konfiguriert ist, sollten Sie Shellzugriff auf das ferne System erhalten. Der Anmeldedialog, in dem Sie zur Eingabe eines Benutzernamens aufgefordert werden, sollte nicht angezeigt werden. Geben Sie `exit` ein, um sich von dieser Test-Shell abzumelden.

5. Legen Sie die richtige Blockgröße für die Bändeinheit fest. Im Folgenden sind die empfohlenen Werte beschrieben:

Eintrag	Beschreibung
Datenträger mit 9 Spuren bzw. 0,25-Zoll-Datenträger:	512
8-mm- und 4-mm-Datenträger:	1024

Wenn Sie unsicher sind und die aktuelle Blockgröße der Bändeinheit überprüfen möchten, verwenden Sie den Befehl `tctl`. Beispiel:

```
tctl -f /dev/rmt0 status
```

Wenn Sie die Blockgröße für die Bändeinheit ändern möchten, verwenden Sie den Befehl `chdev`. Beispiel:

```
chdev -l rmt0 -a block_size=1024
```

6. Erstellen Sie mit einer der folgenden Methoden Ihr Archiv:

Sicherung nach Namen

Verwenden Sie den folgenden Befehl, um über Remote-Zugriff ein Sicherungsarchiv nach Namen zu erstellen:

```
find Pfadname -print | backup -ivqf- | rsh ferner_Host \  
"dd of=/dev/rmt0 bs=Blockgröße conv=sync"
```

Sicherung nach I-Nodes

Wenn Sie über Remote-Zugriff ein Sicherungsarchiv nach I-Nodes erstellen möchten, müssen Sie zuerst Ihr Dateisystem abhängen und anschließend den Befehl `backup` verwenden. Beispiel:

```
umount /myfs  
backup -0 -uf- /myfs | rsh ferner_Host \  
"dd of=/dev/rmt0 bs=Blockgröße conv=sync"
```

Ein Archiv erstellen und auf ein fernes Band kopieren

Verwenden Sie den folgenden Befehl, um ein Archiv zu erstellen und auf die ferne Bändeinheit zu kopieren:

```
find Pfadname -print | cpio -ovcB | rsh ferner_Host \  
"dd ibs=5120 obs=Blockgröße of=/dev/rmt0"
```


Ein tar-Archiv erstellen

Verwenden Sie den folgenden Befehl, um über Remote-Zugriff ein **tar**-Archiv zu erstellen:

```
tar -cvdf- Pfadname | rsh ferner_Host \  
"dd of=/dev/rmt0 bs=Blockgröße conv=sync"
```

Eine ferne Sicherung erstellen

Verwenden Sie den folgenden Befehl, um eine ferne Sicherung des Dateisystems `/myfs` zu erstellen:

```
rdump -u -0 -f ferner_Host:/dev/rmt0 /myfs
```

Mit dem Flag `-u` wird das System angewiesen, die Datensätze mit der aktuellen Sicherungsstufe in der Datei `/etc/dumpdates` zu aktualisieren. `-0` ist die Einstellung des Flags für die *Sicherungsstufe*. Die Sicherungsstufe 0 gibt an, dass alle Dateien im Verzeichnis `/myfs` gesichert werden sollen. Weitere Informationen finden Sie in der Beschreibung des Befehls **rdump** in der Veröffentlichung *Commands Reference, Volume 4*.

7. Für das Wiederherstellen Ihres fernen Archivs können Sie eine der folgenden Methoden verwenden:

Eine Sicherung nach Namen wiederherstellen

Verwenden Sie den folgenden Befehl, um ein fernes Sicherungsarchiv nach Namen wiederherzustellen:

```
rsh ferner_Host "dd if=/dev/rmt0 bs=Blockgröße" | restore \  
-xvqdf- Pfadname
```

Eine Sicherung nach I-Nodes wiederherstellen

Verwenden Sie den folgenden Befehl, um ein fernes Sicherungsarchiv nach I-Nodes wiederherzustellen:

```
rsh ferner_Host "dd if=/dev/rmt0 bs=Blockgröße" | restore \  
-xvqf- Pfadname
```

Ein fernes cpio-Archiv wiederherstellen

Verwenden Sie den folgenden Befehl, um ein fernes Archiv wiederherzustellen, das mit dem Befehl **cpio** erstellt wurde:

```
rsh ferner_Host "dd if=/dev/rmt0 ibs=Blockgröße obs=5120" | \  
cpio -icvdumB
```

Ein tar-Archiv wiederherstellen

Verwenden Sie den folgenden Befehl, um ein fernes **tar**-Archiv wiederherzustellen:

```
rsh ferner_Host "dd if=/dev/rmt0 bs=Blockgröße" | tar -xvpf- Pfadname
```

Eine ferne Sicherung wiederherstellen

Verwenden Sie den folgenden Befehl, um eine ferne Sicherung des Dateisystems `/myfs` wiederherzustellen:

```
cd /myfs  
rrestore -rvf ferner_Host:/dev/rmt0
```

Benutzerdateien aus einem Sicherungsimage wiederherstellen:

Wenn Sie ein versehentlich gelöscht Sicherungsimage wiederherstellen müssen, ist die größte Schwierigkeit festzustellen, welches der Sicherungsbänder diese Datei enthält. Mit dem Befehl **restore -T** können Sie den Inhalt eines Archivs auflisten. Es empfiehlt sich, die Datei im Verzeichnis `/tmp` wiederherzustellen, damit Sie die anderen Dateien des Benutzers nicht versehentlich überschreiben.

Stellen Sie sicher, dass die Einheit angeschlossen und verfügbar ist. Geben Sie zum Überprüfen der Verfügbarkeit Folgendes ein:

```
lsdev -C | pg
```

Wenn zur Sicherungsstrategie inkrementelle Sicherungen gehören, ist es hilfreich beim Benutzer nachzufragen, wann die Datei zuletzt geändert wurde. Diese Angabe hilft Ihnen festzustellen, welche inkremen-

tellesicherung die Datei enthält. Wenn Sie diese Informationen nicht abrufen können oder die gefundenen Informationen nicht korrekt sind, durchsuchen Sie die inkrementellen Sicherungen in umgekehrter Reihenfolge (7, 6, 5, ...). Für inkrementelle Dateisystemsicherungen ist das Flag **-i** (Dialogmodus) des Befehls **restore** sehr hilfreich, um verloren gegangene Dateien zu suchen und wiederherzustellen. (Der Dialogmodus ist auch hilfreich für das Wiederherstellen eines einzelnen Benutzeraccounts aus einer Sicherung des Dateisystems /home.)

Die Prozeduren in der folgenden Tabelle beschreiben, wie eine Wiederherstellung der Stufe 0 (vollständig) eines Verzeichnisses oder Dateisystems implementiert wird.

Tasks für die Wiederherstellung über ein Sicherungsimage		
Task	SMIT-Direktaufruf	Befehl oder Datei
Einzelne Benutzerdateien wiederherstellen	smit restfile	Weitere Informationen finden Sie in der Beschreibung des Befehls restore .
Ein Benutzerdateisystem wiederherstellen	smit restfilesys	<ol style="list-style-type: none"> 1. mkfs /dev/hd1 2. mount /dev/hd1 /filesys 3. cd /filesys 4. restore -r
Eine Benutzerdatenträgergruppe wiederherstellen	smit restvg	Weitere Informationen finden Sie in der Beschreibung des Befehls restvg -q .

Zugehörige Tasks:

„Ein beschädigtes Dateisystem korrigieren“ auf Seite 476

Dateisysteme können beschädigt werden, wenn die I-Node- oder Superblockinformationen für die Verzeichnisstruktur des Dateisystems beschädigt werden.

Zugriff auf eine nicht verlinkte oder gelöschte Systembibliothek wiederherstellen:

Falls die vorhandene Bibliothek **libc.a** nicht verfügbar ist, werden die meisten Betriebssystembefehle nicht erkannt.

Die wahrscheinlichsten Ursachen für dieses Problem sind im Folgenden aufgeführt:

- Die Verbindung in `/usr/lib` ist nicht mehr vorhanden.
- Die Datei in `/usr/ccs/lib` wurde gelöscht.

Die folgende Prozedur beschreibt, wie Sie den Zugriff auf die Bibliothek **libc.a** wiederherstellen können. Damit Sie diese Prozedur ausführen können, müssen Sie das System herunterfahren. Sie sollten für diese Prozedur einen Zeitpunkt wählen, der so wenig wie möglich Auswirkungen auf Ihren Arbeitsablauf hat.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

Zugehörige Informationen:

mount command

unmount command

reboot command

Eine gelöschte symbolische Verbindung wiederherstellen:

Gehen Sie wie folgt vor, um eine symbolische Verbindung zwischen der Bibliothek `/usr/lib/libc.a` und dem Pfad `/usr/ccs/lib/libc.a` wiederherzustellen:

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Melden Sie sich als Benutzer mit Rootberechtigung an, und setzen Sie die Umgebungsvariable **LIBPATH** mit den folgenden Befehlen auf das Verzeichnis `/usr/ccs/lib`:

```
# LIBPATH=/usr/ccs/lib:/usr/lib
# export LIBPATH
```

Danach sollte es möglich sein, Systembefehle auszuführen.

2. Wenn Sie die Verbindung zwischen der Bibliothek `/usr/lib/libc.a` und dem Verzeichnis `/lib` zum Verzeichnis `/usr/lib` wiederherstellen möchten, geben Sie die folgenden Befehle ein:

```
ln -s /usr/ccs/lib/libc.a /usr/lib/libc.a
ln -s /usr/lib /lib
```

Danach sollten Sie die Befehle wie gewohnt ausführen können. Sollten Sie immer noch nicht auf eine Shell zugreifen können, überspringen Sie den Rest der Prozedur, und fahren Sie mit dem nächsten Schritt, „Eine gelöschte Systembibliotheksdatei wiederherstellen“, fort.

3. Geben Sie den folgenden Befehl ein, um die Definition der Umgebungsvariablen **LIBPATH** aufzuheben:

```
unset LIBPATH
```

Eine gelöschte Systembibliotheksdatei wiederherstellen:

Für die folgende Prozedur, mit der Sie eine gelöschte Systembibliotheksdatei wiederherstellen können, muss das System heruntergefahren werden. Das System wird gebootet, und anschließend wird die Bibliothek von einem aktuellen **mksysb**-Band wiederhergestellt.

1. Vergewissern Sie sich vor dem Warmstart, dass das Feld **PROMPT** in der Datei `bosinst.data` den Wert `yes` hat.
2. Legen Sie das aktuelle **mksysb**-Band in das Bandlaufwerk ein. Das **mksysb** muss dasselbe Betriebssystem mit demselben Wartungspaket bzw. Technology Level wie das installierte System enthalten. Wenn Sie eine Bibliothek `libc.a` aus einem **mksysb** wiederherstellen, die mit der Version auf dem installierten System in Konflikt steht, können Sie keine Befehle absetzen.
3. Führen Sie einen Warmstart durch.
4. Wenn die Anzeige mit den Symbolen erscheint oder wenn Sie einen doppelten Signalton hören, drücken Sie die Taste `F1` so oft, bis das Menü "System Management Services" erscheint.
5. Wählen Sie **Multiboot** aus.
6. Wählen Sie **Installieren von** aus.
7. Wählen Sie die Bandeinheit mit dem **mksysb**-Image und anschließend die Option **Installieren** aus. Es kann mehrere Minuten dauern, bis die nächste Systemanfrage erscheint.
8. Definieren Sie Ihr aktuelles System als Systemkonsole. Drücken Sie dazu die Taste `F1` und anschließend die Eingabetaste.
9. Wählen Sie die Nummer der von Ihnen bevorzugten Sprache aus, und drücken Sie anschließend die Eingabetaste.
10. Wählen Sie **Wartungsmodus für Wiederherstellung des Systembetriebs starten** aus. Geben Sie dazu eine `3` ein, und drücken Sie anschließend die Eingabetaste.
11. Wählen Sie die Option **Auf eine Stammdatenträgergruppe (rootvg) zugreifen** aus. Daraufhin erscheint eine Nachricht, die Ihnen mitteilt, dass Sie ohne erneutes Booten des Systems nicht in die Installationsmenüs zurückkehren können, falls Sie die Stammdatenträgergruppe an diesem Punkt ändern.
12. Geben Sie `0` ein, und drücken Sie die Eingabetaste.
13. Geben Sie die Nummer der gewünschten Datenträgergruppe aus der Liste ein, und drücken Sie anschließend die Eingabetaste.
14. Wählen Sie **Auf diese Datenträgergruppe zugreifen** aus. Geben Sie dazu eine `2` ein, und drücken Sie anschließend die Eingabetaste.
15. Hängen Sie das Stammdatensystem (`/`) und das Dateisystem `/usr` mit den folgenden Befehlen an:

```
mount /dev/hd4 /mnt
mount /dev/hd2 /mnt/usr
cd /mnt
```

16. Geben Sie den folgenden Befehl ein, um die symbolische Verbindung für die Bibliothek `libc.a` (sofern erforderlich) wiederherzustellen:

```
ln -s /usr/ccs/lib/libc.a /mnt/usr/lib/libc.a
```

Führen Sie nach der Befehlsausführung eine der folgenden Aktionen aus:

- Wenn der Befehl erfolgreich war, fahren Sie mit Schritt 20 fort.
- Wenn eine Nachricht erscheint, die Ihnen mitteilt, dass die Verbindung bereits vorhanden ist, fahren Sie mit Schritt 17 fort.

17. Legen Sie die Blockgröße für das Bandlaufwerk mit den folgenden Befehlen fest. In den Befehlen steht `X` für die Nummer des entsprechenden Bandlaufwerks.

```
tctl -f /dev/rmtX rewind
tctl -f /dev/rmtX.1 fsf 1
restbyname -xvqf /dev/rmtX.1 ./tapeblksz
cat tapeblksz
```

Wenn der vom Befehl `cat tapeblksz` zurückgegebene Wert *ungleich* 512 ist, geben Sie die folgenden Befehle ein, und ersetzen Sie `Y` durch den vom Befehl `cat tapeblksz` zurückgegebenen Wert:

```
ln -sf /mnt/usr/lib/methods /etc/methods
/etc/methods/chgdevn -l rmtX -a block_size=Y
```

Es sollte eine Nachricht angezeigt werden, die Ihnen mitteilt, dass `rmtX` geändert wurde.

18. Vergewissern Sie sich, dass sich das Band an der korrekten Position für die Wiederherstellung der Bibliothek befindet. Geben Sie dazu die folgenden Befehle ein (`X` steht für die Nummer des entsprechenden Bandlaufwerks):

```
tctl -f /dev/rmtX rewind
tctl -f /dev/rmtX.1 fsf 3
```

19. Stellen Sie die fehlende Bibliothek mit einem der folgenden Befehle wieder her (`X` steht für die Nummer des entsprechenden Bandlaufwerks):

- Wenn Sie nur die Bibliothek `libc.a` wiederherstellen möchten, geben Sie den folgenden Befehl ein:
`restbyname -xvqf /dev/rmtX.1 ./usr/ccs/lib/libc.a`
- Wenn Sie das Verzeichnis `/usr/ccs/lib` wiederherstellen möchten, geben Sie den folgenden Befehl ein:
`restbyname -xvqf /dev/rmtX.1 ./usr/ccs/lib`
- Wenn Sie das Verzeichnis `/usr/ccs/bin` wiederherstellen möchten, geben Sie den folgenden Befehl ein:
`restbyname -xvqf /dev/rmtX.1 ./usr/ccs/bin`

20. Schreiben Sie die Daten mit den folgenden Befehlen auf die Platte:

```
cd /mnt/usr/sbin
./sync;./sync;./sync
```

21. Hängen Sie das Dateisystem `/usr` und das Stammdateisystem (`/`) mit den folgenden Befehlen ab:

```
cd /
umount /dev/hd2
umount /dev/hd4
```

Wenn einer der Befehle `umount` scheitert, schalten Sie das System aus und wieder ein, und wiederholen Sie die Prozedur.

22. Führen Sie mit dem folgenden Befehl einen Warmstart durch:

```
reboot
```

Nach dem Warmstart des Systems sollten die Betriebssystembefehle wieder verfügbar sein.

Ein beschädigtes Boot-Image erneut erstellen:

Im Folgenden wird beschrieben, wie Sie ein beschädigtes Boot-Image identifizieren und erneut erstellen.

Wenn Ihre Maschine eingeschaltet ist und Sie wissen, dass das Boot-Image beschädigt ist oder gelöscht wurde, können Sie das Boot-Image mit dem Befehl **bosboot** erneut erstellen. Sie müssen diesen Befehl als Root ausführen.

Achtung: Führen Sie niemals einen Warmstart durch, wenn Sie vermuten, dass das Boot-Image beschädigt ist.

In den folgenden Prozeduren wird davon ausgegangen, dass das System aufgrund eines beschädigten Boot-Image nicht mehr ordnungsgemäß bootet. Um Ihr System vor einem potenziellen Daten- oder Funktionalitätsverlust zu schützen, sollten Sie für das Herunterfahren des Systems einen Zeitpunkt wählen, der Ihren Arbeitsablauf am wenigsten stört.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Legen Sie den Produktdatenträger in das Laufwerk ein.
2. Schalten Sie die Maschine ein. Befolgen Sie hierbei die Anleitungen, die Sie zusammen mit Ihrem System erhalten haben.
3. Wählen Sie im Menü **System Management Services** die Option **Multiboot** aus.
4. Wählen Sie in der nächsten Anzeige die Option **Installieren von** aus.
5. Wählen Sie die Einheit mit dem eingelegten Produktdatenträger aus, und wählen Sie dann **Installieren** aus.
6. Wählen Sie das AIX-Versionssymbol aus.
7. Folgen Sie den Anweisungen, bis Sie die Anzeige erreichen, in der Sie den Installationsmodus auswählen können. Wählen Sie in dieser Anzeige **Wartungsmodus für Wiederherstellung des Systembetriebs starten** aus.
8. Wählen Sie die Option **Auf eine Stammdatenträgergruppe (rootvg) zugreifen** aus.
9. Folgen Sie den Anweisungen, bis Sie die Anzeige erreichen, in der Sie die Option **Auf diese Datenträgergruppe zugreifen und Shell starten** auswählen können.
10. Verwenden Sie den Befehl **bosboot**, um das Boot-Image zu erstellen. Beispiel:

```
bosboot -a -d /dev/hdisk0
```

Falls der Befehl fehlschlägt und die Nachricht

```
0301-165 bosboot: WARNUNG! bosboot ist fehlgeschlagen - Einheit nicht booten.
```

angezeigt wird, können Sie versuchen, den Fehler wie folgt zu beheben, und dann den Befehl **bosboot** erneut ausführen, um ein Boot-Image zu erstellen:

- Löschen Sie den logischen Standardbootdatenträger (hd5), und erstellen Sie dann einen neuen logischen Datenträger (hd5).

ODER

- Führen Sie das Diagnoseprogramm für die Festplatte aus. Reparieren oder tauschen Sie die Platte bei Bedarf aus.

Falls der Befehl **bosboot** weiterhin fehlschlägt, wenden Sie sich an den Kundendienst.

Achtung: Sollte der Befehl **bosboot** beim Erstellen eines Boot-Image fehlschlagen, führen Sie keinen Warmstart durch.

11. Wenn der Befehl **bosboot** erfolgreich ausgeführt wurde, können Sie Ihr System mit dem Befehl **reboot** erneut booten.

Zugehörige Konzepte:

„Systemstart“ auf Seite 4

Wenn das Basisbetriebssystem gestartet wird, leitet das System eine komplexe Folge von Tasks ein. Unter normalen Bedingungen werden diese Tasks automatisch ausgeführt.

Zugehörige Informationen:

bosboot command

Eine Onlinesicherung eines JFS erstellen:

Wenn Sie eine Onlinesicherung eines angehängten Journaled File System (JFS) oder eines erweiterten Journaled File System (JFS2) erstellen, wird ein statisches Image des logischen Datenträgers erstellt, in dem das Dateisystem enthalten ist.

Wenn Sie eine Onlinesicherung eines angehängten JFS erstellen möchten, müssen der logische Datenträger, in dem sich das Dateisystem befindet, und der logische Datenträger, in dem sich das zugehörige Protokoll befindet, gespiegelt werden.

Anmerkung: Da die Dateien im asynchronen Modus geschrieben werden, enthält die abgetrennte Kopie möglicherweise nicht alle Daten, die unmittelbar vor der Abtrennung geschrieben wurden. Alle Änderungen, die nach der Abtrennung vorgenommen werden, sind unter Umständen nicht in der Sicherungskopie enthalten. Deshalb wird empfohlen, alle Aktivitäten im Dateisystem auf ein Minimum zu reduzieren, während die Abtrennung vollzogen wird.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

Geben Sie Folgendes ein, um eine gespiegelte Kopie des Dateisystems /home/xyz abzutrennen und an einen neuen Mountpunkt mit dem Namen /jfsstaticcopy anzuhängen:

```
chfs -a splitcopy=/jfsstaticcopy /home/xyz
```

Mit dem Attribut **copy** können Sie steuern, welche gespiegelte Kopie als Sicherung verwendet wird. Es wird standardmäßig die zweite gespiegelte Kopie verwendet, wenn Sie keine Kopie angeben. Beispiel:

```
chfs -a splitcopy=/jfsstaticcopy -a copy=1 /home/xyz
```

Damit ist eine schreibgeschützte Kopie des Dateisystems in /jfsstaticcopy verfügbar. Alle Änderungen, die nach der Abtrennung der Kopie am Originaldateisystem vorgenommen wurden, sind in der Sicherungskopie nicht enthalten.

Verwenden Sie den folgenden Befehl, um das abgetrennte JFS-Image als gespiegelte Kopie am Mountpunkt /testcopy wieder zu integrieren:

```
rmfs /testcopy
```

Der Befehl **rmfs** setzt den Status (Abgetrennt) der Kopie des Dateisystems zurück und ermöglicht dadurch die erneute Integration als gespiegelte Kopie.

Momentaufnahme eines JFS2 erstellen und sichern:

Sie können eine Momentaufnahme (oder Snapshot) eines angehängten JFS2 erstellen, die ein konsistentes Image des Dateisystems auf Blockebene zu einem bestimmten Zeitpunkt liefert.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

Das Image bleibt stabil, selbst wenn das Dateisystem, das für die Erstellung der Momentaufnahme verwendet wurde, das so genannte *snappedFS*, sich weiter ändert. Die Momentaufnahme hat dieselben Sicherheitsberechtigungen wie das *snappedFS* zu dem Zeitpunkt, als die Momentaufnahme erstellt wurde.

Im folgenden Szenario erstellen Sie mit einem einzigen Befehl eine Momentaufnahme und sichern sie auf einem austauschbaren Datenträger, ohne das Dateisystem abzuhängen oder stillzulegen. Der Befehl ist **backsnap**. Sie können die Momentaufnahme auch für andere Zwecke verwenden, z. B. um auf die Datei- oder Verzeichnisstände zuzugreifen, die zu dem Zeitpunkt aktuell waren, als die Momentaufnahme erstellt wurde. Mit SMIT oder mit den Befehlen **backsnap** und **snapshot** können Sie verschiedene Prozeduren für die Erstellung einer Momentaufnahme ausführen.

Verwenden Sie den folgenden Befehl, um eine Momentaufnahme des Dateisystems `/home/abc/test` zu erstellen und diese auf der Bandeinheit `/dev/rmt0` zu sichern:

```
backsnap -m /tmp/snapshot -s size=16M -i f/dev/rmt0 /home/abc/test
```

Dieser Befehl erstellt einen logischen Datenträger mit einer Speicherkapazität von 16 Megabytes für die Momentaufnahme des JFS2-Dateisystems (`/home/abc/test`). Die Momentaufnahme wird an `/tmp/snapshot` angehängt, und anschließend wird die Momentaufnahme unter ihrem Namen auf der Bandeinheit gesichert. Die Momentaufnahme bleibt nach Abschluss der Sicherung angehängt. Verwenden Sie das Flag **-R** mit dem Befehl **backsnap**, wenn die Momentaufnahme nach Abschluss der Sicherung entfernt werden soll.

Zugehörige Konzepte:

„Dateisysteme“ auf Seite 453

Ein *Dateisystem* ist eine hierarchische Struktur (Dateibaum) von Dateien und Verzeichnissen.

Zugehörige Informationen:

backsnap command

chfs command

rmfs command

snapshot command

Externe Momentaufnahme eines JFS2 erstellen und sichern:

Sie können eine Momentaufnahme (oder Snapshot) eines angehängten JFS2 erstellen, die ein konsistentes Image des Dateisystems auf Blockebene zu einem bestimmten Zeitpunkt liefert.

Das Image bleibt stabil, selbst wenn das Dateisystem, das für die Erstellung der Momentaufnahme verwendet wurde, das so genannte *snappedFS*, sich weiter ändert. Die Momentaufnahme hat dieselben Sicherheitsberechtigungen wie das *snappedFS* zu dem Zeitpunkt, als die Momentaufnahme erstellt wurde.

Im folgenden Szenario verwenden Sie den Befehl **backsnap**, um eine externe Momentaufnahme zu erstellen und diese auf einem austauschbaren Datenträger zu sichern, ohne das Dateisystem abzuhängen oder stillzulegen. Sie können die Momentaufnahme auch für andere Zwecke verwenden, z. B. um auf die Datei- oder Verzeichnisstände zuzugreifen, die zu dem Zeitpunkt aktuell waren, als die Momentaufnahme erstellt wurde. Mit SMIT oder mit den Befehlen **backsnap** und **snapshot** können Sie verschiedene Prozeduren für die Erstellung einer Momentaufnahme ausführen.

Geben Sie den folgenden Befehl ein, um eine externe Momentaufnahme des Dateisystems `/home/abc/test` zu erstellen und diese Momentaufnahme unter ihrem Namen auf der Bandeinheit `/dev/rmt0` zu sichern:

```
backsnap -m /tmp/snapshot -s size=16M -if/dev/rmt0 /home/abc/test
```

Dieser Befehl erstellt einen logischen Datenträger mit einer Speicherkapazität von 16 MB für die Momentaufnahme des JFS2-Dateisystems `/home/abc/test`. Die Momentaufnahme wird an das Verzeichnis `/tmp/snapshot` angehängt. Anschließend wird eine Sicherung der Momentaufnahme unter ihrem Namen auf der Bandeinheit durchgeführt. Nach dem Abschluss der Sicherung wird die Momentaufnahme abge-

hängt, bleibt aber weiterhin verfügbar. Geben Sie das Flag **-R** mit dem Befehl **backsnap** an, wenn die Momentaufnahme nach Fertigstellung der Sicherung entfernt werden soll.

Zugehörige Konzepte:

„Dateisysteme“ auf Seite 453

Ein *Dateisystem* ist eine hierarchische Struktur (Dateibaum) von Dateien und Verzeichnissen.

Interne Momentaufnahme eines JFS2 erstellen und sichern:

Sie können eine Momentaufnahme (oder Snapshot) eines angehängten JFS2 erstellen, die ein konsistentes Image des Dateisystems auf Blockebene zu einem bestimmten Zeitpunkt liefert.

Das Image bleibt stabil, selbst wenn das Dateisystem, das für die Erstellung der Momentaufnahme verwendet wurde, das so genannte *snappedFS*, sich weiter ändert. Die Momentaufnahme hat dieselben Sicherheitsberechtigungen wie das *snappedFS* zu dem Zeitpunkt, als die Momentaufnahme erstellt wurde.

Im folgenden Szenario verwenden Sie den Befehl **backsnap**, um eine interne Momentaufnahme zu erstellen und diese auf einem austauschbaren Datenträger zu sichern, ohne das Dateisystem abzuhängen oder stillzulegen. Sie können die Momentaufnahme auch für andere Zwecke verwenden, z. B. um auf die Datei- oder Verzeichnisstände zuzugreifen, die zu dem Zeitpunkt aktuell waren, als die Momentaufnahme erstellt wurde. Mit SMIT oder mit den Befehlen **backsnap** und **snapshot** können Sie verschiedene Prozeduren für die Erstellung einer Momentaufnahme ausführen.

Geben Sie den folgenden Befehl ein, um eine interne Momentaufnahme des Dateisystems `/home/abc/test` zu erstellen und diese Momentaufnahme unter ihrem Namen auf der Bandeinheit `/dev/rmt0` zu sichern:

```
backsnap -n mysnapshot -if/dev/rmt0 /home/abc/test
```

Dieser Befehl erstellt eine interne Momentaufnahme mit dem Namen `mysnapshot` des Dateisystems `/home/abc/test`. Die Momentaufnahme wird über das Verzeichnis `/home/abc/test/.snapshot/mysnapshot` aufgerufen und anschließend wird eine Sicherung der Momentaufnahme auf der Bandeinheit durchgeführt.

Geben Sie das Flag **-R** mit dem Befehl **backsnap** an, wenn die Momentaufnahme nach Fertigstellung der Sicherung entfernt werden soll.

Zugehörige Konzepte:

„Dateisysteme“ auf Seite 453

Ein *Dateisystem* ist eine hierarchische Struktur (Dateibaum) von Dateien und Verzeichnissen.

Dateien komprimieren (Befehle **compress** und **pack**):

Mit dem Befehl **compress** und dem Befehl **pack** können Sie Dateien zum Speichern komprimieren.

Mit dem Befehl **uncompress** und dem Befehl **unpack** können Sie die wiederhergestellten Dateien dekomprimieren.

Das Komprimieren und Dekomprimieren von Dateien nimmt eine gewisse Zeit in Anspruch, nach dem Packen benötigen die Daten jedoch weniger Speicherplatz auf dem Sicherungsdатenträger.

Sie können zum Komprimieren eines Dateisystems zwischen den folgenden Methoden wählen:

- Flag **-p** mit dem Befehl **backup** verwenden
- Befehl **compress** oder **pack** verwenden

Vorteile der Dateikomprimierung:

- Einsparung von Geld und Zeit durch das Komprimieren der Dateien vor dem Senden über ein Netz
- Einsparung von Speicher- und Archivsystemressourcen:
 - Dateisysteme vor der Sicherung komprimieren, um Speicherplatz auf dem Band einzusparen

- Protokolldateien komprimieren, die von Shell-Scripts erstellt werden, die in der Nacht ausgeführt werden. Die Scripts können auf einfache Weise so eingestellt werden, dass sie vor ihrer Beendigung die Dateien komprimieren.
- Dateien komprimieren, auf die momentan nicht zugegriffen wird. Die Dateien eines Benutzers, der für einen längeren Zeitraum nicht am System arbeitet, können beispielsweise komprimiert und in das **tar**-Archiv auf der Platte oder einem Band gestellt und bei Bedarf wiederhergestellt werden.

Anmerkung:

- Beim Komprimieren reicht der Arbeitsbereich im Dateisystem für den Befehl **compress** möglicherweise nicht aus. Mit dem Befehl werden zunächst die komprimierten Dateien erstellt, bevor eine der nicht komprimierten Dateien gelöscht wird; dadurch wird etwa 50 % mehr Speicherplatz als die Gesamtgröße der Dateien benötigt.
- Das Komprimieren einer Datei wird möglicherweise abgebrochen, da sie bereits komprimiert ist. Die Ausführung des Befehls **compress** wird abgebrochen, wenn die Dateien nicht verkleinert werden können.

Einzelheiten zu den Rückgabewerten finden Sie in der Beschreibung des Befehls **compress** in der Veröffentlichung "Commands Reference". Im Allgemeinen lassen sich die Probleme, die beim Komprimieren von Dateien auftreten, jedoch wie folgt zusammenfassen:

- Beim Komprimieren reicht der Arbeitsbereich im Dateisystem für den Befehl möglicherweise nicht aus. Da der Befehl **compress** die komprimierten Dateien erstellt, bevor er die nicht komprimierten Dateien löscht, benötigt er für jede zu komprimierende Datei zwischen 50 % und 100 % der Dateigröße mehr Plattenspeicherplatz.
- Das Komprimieren einer Datei wird möglicherweise abgebrochen, da sie bereits komprimiert ist. Wenn der Befehl **compress** die Dateigröße nicht verringern kann, schlägt er fehl.

Dateien mit dem Befehl compress komprimieren:

Mit dem Befehl **compress** wird die Größe von Dateien mit der angepassten Lempel-Zev-Codierung verringert.

Jede im Parameter *Datei* angegebene Originaldatei wird durch eine komprimierte Datei ersetzt, an deren Name ein *.Z* angefügt wird. Die komprimierte Datei ist demselben Eigner zugeordnet und übernimmt die Modi, Zugriffs- und Änderungszeiten der Originaldatei. Wenn keine Dateien angegeben werden, wird die Standardeingabe in die Standardausgabe komprimiert. Falls sich die Größe der Datei durch das Komprimieren nicht verringert, wird eine Nachricht in die Standardfehlerausgabe geschrieben. Die Originaldatei wird nicht ersetzt.

Mit dem Befehl **uncompress** können Sie komprimierte Dateien im Originalformat wiederherstellen.

Das Komprimierungsergebnis hängt von der Größe der Eingabe, den angegebenen *Bits* pro Code sowie der Verteilung allgemeiner Unterzeichenfolgen ab. Quellcode oder englischer Text wird in der Regel um 50 - 60 % komprimiert. Die Komprimierung mit dem Befehl **compress** ist im Allgemeinen kompakter und erfordert weniger Zeit als die Komprimierung mit dem Befehl **pack**, bei dem die angepasste Huffman-Codierung verwendet wird.

Geben Sie beispielsweise Folgendes ein, um die Datei *foo* zu komprimieren und das Ergebnis der Komprimierung in Prozent in die Standardfehlerausgabe zu schreiben:

```
compress -v foo
```

Die vollständige Syntax des Befehls **compress** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

Dateien mit dem Befehl pack komprimieren:

Mit dem Befehl **pack** werden die im Parameter *Datei* angegebenen Dateien in komprimierter Form nach der Huffman-Codierung gespeichert.

Die Eingabedatei wird durch die gepackte Datei ersetzt, deren Name vom Namen der Originaldatei abgeleitet ist (*Datei.z*). Die Zugriffsmodi, Zugriffs- und Änderungsdaten sowie der Eigner bleiben gegenüber der Originaldatei unverändert. Der Name der Eingabedatei darf nicht mehr als 253 Byte enthalten, damit noch Platz für das angefügte Suffix *.z* bleibt. Wurde der Befehl **pack** erfolgreich ausgeführt, wird die Originaldatei gelöscht.

Mit dem Befehl **unpack** können Sie gepackte Dateien im Originalformat wiederherstellen.

Kann mit dem Befehl **pack** keine kleinere Datei erstellt werden, wird die Verarbeitung abgebrochen und die Nachricht angezeigt, dass kein Speicherplatz eingespart werden kann. (Im Allgemeinen tritt dieser Fall bei kleinen Dateien oder bei Dateien mit gleichmäßiger Zeichenverteilung auf.) Die Größe des eingesparten Speicherplatzes hängt von der Größe der Eingabedatei und der Häufigkeitsverteilung der Zeichen ab. Da jede Datei *.z* mit einer Entschlüsselungsbaumstruktur beginnt, kann mit Dateien, die kleiner als drei Blöcke sind, kein Speicherplatz eingespart werden. In der Regel werden Textdateien um 25 - 40 % verkleinert.

Der Rückkehrcode des Befehls **pack** ist die Anzahl der Dateien, die nicht gepackt werden konnten. Dateien werden nicht gepackt, wenn eine der folgenden Bedingungen zutrifft:

- Die Datei ist bereits gepackt.
- Der Name der Eingabedatei ist größer als 253 Bytes.
- Die Datei besitzt Verbindungen.
- Die Datei ist ein Verzeichnis.
- Die Datei kann nicht geöffnet werden.
- Durch das Packen werden keine Speicherblöcke eingespart.
- Es ist bereits eine Datei mit dem Namen *Datei.z* vorhanden.
- Die Datei *.z* kann nicht erstellt werden.
- Während der Verarbeitung trat ein E/A-Fehler auf.

Geben Sie beispielsweise Folgendes ein, um die Dateien *chap1* und *chap2* zu komprimieren:

```
pack chap1 chap2
```

Damit werden die Dateien *chap1* und *chap2* komprimiert und durch die Dateien *chap1.z* und *chap2.z* ersetzt. Der Befehl **pack** zeigt den Prozentsatz der Verkleinerung für jede Datei an.

Die vollständige Syntax des Befehls **pack** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

Komprimierte Dateien dekomprimieren (Befehle uncompress und unpack):

Mit den Befehlen **uncompress** und **unpack** können Sie komprimierte Dateien dekomprimieren.

Dateien mit dem Befehl uncompress dekomprimieren

Mit dem Befehl **uncompress** können Sie Originaldateien, die mit dem Befehl **compress** komprimiert wurden, wiederherstellen. Jede der mit der Variablen *Datei* angegebenen komprimierten Dateien wird gelöscht und durch eine dekomprimierte Kopie ersetzt. Die dekomprimierte Datei hat denselben Namen wie die komprimierte Version, allerdings ohne die Erweiterung *.Z*. Der Eignerstatus, die Zugriffsmodi sowie Zugriffs- und Änderungsdatum bleiben gegenüber der Originaldatei unverändert. Wenn keine Dateien angegeben sind, wird die Standardeingabe in der Standardausgabe erweitert.

Der Befehl **zcat** gleicht im Wesentlichen dem Befehl **uncompress**, allerdings schreibt er die dekomprimierte Ausgabe immer in die Standardausgabe.

Geben Sie beispielsweise Folgendes ein, um die Komprimierung der Datei `foo` aufzuheben:

```
uncompress foo
```

Die vollständige Syntax können Sie der Beschreibung des Befehls **uncompress** in der Veröffentlichung *Commands Reference, Volume 5* entnehmen.

Dateien mit dem Befehl **unpack** dekomprimieren

Der Befehl **unpack** dekomprimiert Dateien, die mit dem Befehl **pack** erstellt wurden. Der Befehl **unpack** sucht für jede angegebene Datei eine Datei mit dem Namen `Datei.z`. Wenn diese Datei eine gepackte Datei ist, ersetzt der Befehl **unpack** sie durch die nicht komprimierte Version. Der Befehl **unpack** benennt die neue Datei um, indem er das Suffix `.z` von `Datei` entfernt. Die neue Datei wird vom Befehl **unpack** wieder mit dem ursprünglichen Namen benannt, indem das Suffix `.z` vom Namen der Datei entfernt wird. Die Zugriffsmodi, Zugriffs- und Änderungsdaten sowie der Eignerstatus bleiben gegenüber der ursprünglichen gepackten Datei unverändert.

Der Befehl **unpack** bearbeitet nur solche Dateien, die mit `.z` enden. Wenn Sie einen Dateinamen angeben, der nicht mit `.z` endet, fügt der Befehl **unpack** das Suffix hinzu und sucht im Verzeichnis einen Dateinamen mit diesem Suffix.

Der Exit-Wert ist die Anzahl der Dateien, die der Befehl **unpack** nicht entpacken konnte. Eine Datei kann in folgenden Fällen nicht entpackt werden:

- Der Dateiname (ohne die Erweiterung `.z`) enthält mehr als 253 Bytes.
- Die Datei kann nicht geöffnet werden.
- Die Datei ist nicht gepackt.
- Es ist bereits eine Datei mit dem Namen der nicht gepackten Datei vorhanden.
- Die entpackte Datei kann nicht erstellt werden.

Anmerkung: Der Befehl **unpack** schreibt eine Warnung in die Standardfehlerausgabe, wenn die zu entpackende Datei Verbindungen hat. Die neue entpackte Datei hat eine andere I-Node-Nummer als die gepackte Datei, aus der sie erstellt wurde. Alle anderen mit der ursprünglichen I-Node-Nummer verknüpften Dateien bleiben im gepackten Zustand bestehen.

Wenn Sie beispielsweise die gepackten Dateien `chap1.z` und `chap2.z` entpacken möchten, geben Sie Folgendes ein:

```
unpack chap1.z chap2
```

Dieser Befehl dekomprimiert die gepackten Dateien `chap1.z` und `chap2.z` und ersetzt sie durch die Dateien `chap1` und `chap2`.

Anmerkung: Sie können mit dem Befehl **unpack** Dateinamen mit oder ohne das Suffix `.z` angeben.

Die vollständige Syntax können Sie der Beschreibung des Befehls **unpack** in der Veröffentlichung *Commands Reference, Volume 5* entnehmen.

Sicherung von Systemimages und benutzerdefinierten Datenträgergruppen

Die Stammdatenträgergruppe (`rootvg`) ist auf einer Festplatte oder einer Gruppe von Festplatten gespeichert und enthält Startdateien, das Basisbetriebssystem (BOS, Base Operating System), Konfigurationsdaten und alle optionalen Softwareprodukte. Eine *benutzerdefinierte Datenträgergruppe* (auch *Nicht-rootvg-Datenträgergruppe* genannt) enthält normalerweise Datendateien und Anwendungssoftware.

Sie können ein Image des Systems und der Datenträgergruppe mit SMIT oder Befehlsprozeduren sichern. Ein Sicherungsimagedient zwei Zwecken. Zum einen kann mit dem Systemsicherungsimaged ein beschä-

digtes System wiederhergestellt werden. Zum anderen kann mit dem Sicherungsbild installierte und konfigurierte Software von einem System auf ein anderes übertragen werden.

Die SMIT-Prozeduren verwenden den Befehl **mksysb**, um ein Sicherungsbild zu erstellen, das auf Band oder in einer Datei gespeichert werden kann. Wenn Sie das Band wählen, schreibt das Sicherungsprogramm ein *Boot-Image* auf das Band, das dann für Installationen verwendet werden kann.

Anmerkung:

- Auf einem PowerPC-basierten Personal Computer können keine Startbänder erstellt, noch können sie zum Starten eines solchen Personal Computer verwendet werden.
- Wenn Sie die SMIT-Methode für die Sicherung wählen, müssen Sie zuerst die Dateigruppe `sysbr` aus dem Softwarepaket `bos.sysmgt` installieren.

Zugehörige Konzepte:

„Sicherungen“ auf Seite 27

Im Allgemeinen werden Sicherungen von Benutzer- und Systemdaten aufbewahrt für den Fall, dass Daten versehentlich entfernt werden oder dass eine Platte defekt ist. Sicherungen lassen sich einfacher verwalten, wenn Benutzerdaten von Systemdaten getrennt aufbewahrt werden.

Zugehörige Informationen:

Installing optional software products and service updates

Systemimage und benutzerdefinierte Datenträgergruppen sichern:

Sie können Sicherungen des Systemimage und der benutzerdefinierten Datenträgergruppen erstellen.

Bevor Sie die Datenträgergruppe `rootvg` sichern, müssen Sie Folgendes sicherstellen:

- Die gesamte Hardware, einschließlich externer Einheiten, wie z. B. Band- und CD-ROM-Laufwerken, muss installiert sein.
- Diese Sicherungsprozedur erfordert die Dateigruppe `sysbr`, die sich im Softwarepaket mit den Systemverwaltungstools und Anwendungen für das Basisbetriebssystem befindet. Geben Sie den folgenden Befehl ein, um festzustellen, ob die Dateigruppe `sysbr` auf Ihrem System installiert ist:

```
lslpp -l bos.sysmgt.sysbr
```

Wenn die Dateigruppe `sysbr` auf Ihrem System installiert ist, setzen Sie die Sicherungsprozeduren fort. Listet der Befehl **lslpp** die Dateigruppe `sysbr` nicht auf, müssen Sie die Dateigruppe installieren, bevor Sie die Sicherungsprozedur fortsetzen.

```
installp -agqXd Einheit bos.sysmgt.sysbr
```

Einheit steht hier für die Position der Software, z. B. `/dev/rmt0` für ein Bandlaufwerk.

Bevor Sie eine benutzerdefinierte Datenträgergruppe sichern, müssen Sie Folgendes sicherstellen:

- Vor der Sicherung muss die Datenträgergruppe aktiviert werden, und die Dateisysteme müssen angehängt werden.

Achtung: Bei der Ausführung des Befehls **savevg** geht das gesamte, zuvor auf dem ausgewählten Ausgabedatenträger gespeicherte Material verloren.

- Stellen Sie sicher, dass seit der letzten Reinigung der Sicherungseinheit nicht zu viel Zeit vergangen ist, um Fehler zu verhindern.

Die folgenden Prozeduren beschreiben, wie ein installierbares Image des Systems erstellt wird.

Systemtasks sichern

<i>Task</i>	<i>SMIT-Direktaufruf</i>	<i>Befehl oder Datei</i>
Datenträgergruppe rootvg sichern	<ol style="list-style-type: none"> 1. Melden Sie sich als Root an. 2. Hängen Sie die Dateisysteme für die Sicherung an.¹ smit mountfs 3. Hängen Sie alle lokalen Verzeichnisse ab, die über ein anderes lokales Verzeichnis angehängt sind. smit umountfs 4. Stellen Sie sicher, dass mindestens 8,8 MB freier Plattenspeicherplatz im Verzeichnis /tmp verfügbar sind.² 5. Führen Sie die Sicherung durch: smit mksysb 6. Setzen Sie den Schreibschutz für den Sicherungsdatenträger. 7. Notieren Sie alle gesicherten Root- und Benutzerkennwörter. 	<ol style="list-style-type: none"> 1. Melden Sie sich als Root an. 2. Hängen Sie die Dateisysteme für die Sicherung an.¹ Siehe die Beschreibung des Befehls mount. 3. Hängen Sie alle lokalen Verzeichnisse ab, die über ein anderes lokales Verzeichnis angehängt sind. Weitere Informationen finden Sie in der Beschreibung des Befehls umount. 4. Stellen Sie sicher, dass mindestens 8,8 MB freier Plattenspeicherplatz im Verzeichnis /tmp verfügbar sind.² 5. Führen Sie die Sicherung durch. Weitere Informationen finden Sie in der Beschreibung des Befehls mksysb. 6. Setzen Sie den Schreibschutz für den Sicherungsdatenträger. 7. Notieren Sie alle gesicherten Root- und Benutzerkennwörter.
Sicherungsband prüfen ³	smit lsmksysb	
Benutzerdefinierte Datenträgergruppe sichern ⁴	smit savevg	<ol style="list-style-type: none"> 1. Ändern Sie bei Bedarf die Dateisystemgröße, bevor Sie die Sicherung durchführen.⁵ Verwenden Sie den Befehl mkvgdata Name_der_Datenträgergruppe. Editieren Sie anschließend die Datei /tmp/vgdata/<i>Name_der_Datenträgergruppe</i>/<i>Name_der_Datenträgergruppe</i>.data. 2. Speichern Sie die Datenträgergruppe. Weitere Informationen finden Sie in der Beschreibung des Befehls savevg in der Veröffentlichung "Commands Reference".

Anmerkung:

1. Der Befehl **mksysb** sichert keine Dateisysteme, die über NFS (Network File System) angehängt sind.
2. Der Befehl **mksysb** benötigt diesen Arbeitsbereich für die Dauer der Sicherung. Verwenden Sie den Befehl **df**, der Einheiten von 512-Byte-Blöcken zurückmeldet, um den freien Speicherplatz im Verzeichnis /tmp zu bestimmen. Verwenden Sie den Befehl **chfs**, um bei Bedarf die Größe des Dateisystems zu ändern.
3. Diese Prozedur listet den Inhalt eines **mksysb**-Sicherungsbands auf. Anhand der Inhaltsliste kann der größte Teil der Informationen auf dem Band geprüft werden. Allerdings kann nicht verifiziert werden, ob das Band für Installationen gebootet werden kann. Die einzige Möglichkeit sicherzustellen, dass das Boot-Image auf einem **mksysb**-Band ordnungsgemäß funktioniert, besteht darin, vom Band zu booten.
4. Wenn Sie Dateien in einer benutzerdefinierten Datenträgergruppe vom Sicherungsimage ausschließen möchten, erstellen Sie eine Datei mit dem Namen /etc/exclude.*Name_der_Datenträgergruppe*, wobei *Name_der_Datenträgergruppe* für den Namen der Datenträgergruppe steht, die Sie sichern möchten. Editieren Sie anschließend /etc/exclude.*Name_der_Datenträgergruppe*, und geben Sie die Muster für die Dateinamen ein, die nicht in das Sicherungsimage aufgenommen werden sollen. Die Muster in dieser Datei sind Eingaben für die Konventionen für Mustererkennung des Befehls **grep**, auf deren Basis bestimmt wird, welche Dateien von der Sicherung ausgeschlossen werden.
5. Wenn Sie die Datei *Name_der_Datenträgergruppe*.data modifizieren möchten, um die Größe eines Dateisystems zu ändern, dürfen Sie das Flag **-i** bzw. das Flag **-m** nicht mit dem Befehl **savevg** angeben, da die Datei *Name_der_Datenträgergruppe*.data überschrieben wird.

Zugehörige Informationen:

Installing optional software products and service updates
Installing system backups

Konfiguration vor dem Durchführen einer Sicherung:

Konfigurieren Sie das Quellensystem, bevor Sie ein Sicherungsbild des Systems erstellen. Wenn Sie jedoch planen, ein Sicherungsbild für die Installation weiterer, anders konfigurierter Zielsysteme zu verwenden, erstellen Sie das Image, *bevor* Sie das Quellensystem konfigurieren.

Das *Quellensystem* ist das System, von dem Sie die Sicherungskopie erstellen. Das *Zielsystem* ist das System, auf dem Sie die Sicherungskopie installieren.

Das Installationsprogramm installiert automatisch nur die Einheitenunterstützung, die für die Hardwarekonfiguration der installierten Maschine erforderlich ist. Wenn Sie eine Systemsicherung für die Installation anderer Maschinen verwenden, müssen Sie deshalb möglicherweise weitere Einheiten auf dem Quellensystem installieren, bevor Sie das Sicherungsbild erstellen und es zum Installieren eines oder mehrerer Zielsysteme verwenden.

Verwenden Sie den SMIT-Direktaufruf `smit devinst`, um zusätzliche Einheitenunterstützung auf dem Quellensystem zu installieren.

- Wenn ausreichender Plattenspeicherplatz auf dem Quellen- und auf dem Zielsystem verfügbar ist, installieren Sie die Einheitenunterstützung vollständig.
- Wenn der Plattenspeicherplatz auf dem Quellen- und auf dem Zielsystem begrenzt ist, installieren Sie die Einheitenunterstützung selektiv.

Eine Sicherung überträgt die folgenden Konfigurationen vom Quellensystem auf das Zielsystem:

- Informationen zum Paging-Bereich
- Informationen zu logischen Datenträgern
- Informationen zur Stammdatenträgergruppe (`rootvg`)
- Positionen der logischen Partitionen (wenn Sie die Zuordnungsoption ausgewählt haben)

Zugehörige Informationen:

Installing optional software and service updates
Customizing your installation

Dateisysteme anhängen und abhängen:

Bevor Sie eine Sicherung erstellen, müssen Sie alle Dateisysteme anhängen, die Sie sichern möchten, und alle Dateisysteme abhängen, die Sie nicht sichern möchten.

Mit der unter "Sicherungsmethoden" beschriebenen Prozedur werden nur angehängte Dateisysteme in der Stammdatenträgergruppe (`rootvg`) gesichert. Deshalb müssen Sie zunächst alle Dateisysteme anhängen, die Sie sichern möchten. Gleichermaßen müssen Sie die Dateisysteme abhängen, die *nicht* gesichert werden sollen.

Diese Sicherungsprozedur sichert die Dateien zweimal, wenn ein lokales Verzeichnis über ein anderes lokales Verzeichnis in demselben Dateisystem angehängt ist. Wenn Sie beispielsweise `/tmp` über `/usr/tmp` anhängen, werden die Dateien im Verzeichnis `/tmp` zweimal gesichert. Durch diese Duplizierung wird die Anzahl der Dateien, die in einem Dateisystem gespeichert werden können, möglicherweise überschritten. Dies kann dazu führen, dass eine künftige Installation des Sicherungsbilds fehlschlägt.

Sicherheitsaspekte für Sicherungen:

Wenn Sie das Sicherungsbild auf anderen Systemen installieren, möchten Sie möglicherweise aus Sicherheitsgründen nicht, dass Kennwörter und Netzadressen auf die Zielsysteme kopiert werden.

Außerdem werden durch das Kopieren von Netzadressen auf ein Zielsystem Adressen dupliziert, was zu Störungen bei der Netzübertragung führen kann.

Sicherungsimage wiederherstellen:

Wenn Sie das Sicherungsimage wiederherstellen, prüft das System, ob auf dem Zielsystem ausreichend Plattenspeicherplatz verfügbar ist, um alle logischen Datenträger zu erstellen, die in der Sicherung gespeichert sind. Wenn ausreichend Plattenspeicherplatz verfügbar ist, wird die vollständige Sicherung wiederhergestellt. Wenn nicht, wird die Installation angehalten, und das System fordert Sie auf, weitere Ziel-festplatten auszuwählen.

Die auf dem Zielsystem erstellten Dateisysteme haben dieselbe Größe wie auf dem Quellsystem, sofern die Variable **SHRINK** in der Datei `image.data` vor dem Erstellen des Sicherungsimage nicht auf `yes` gesetzt wurde. Eine Ausnahme ist das Verzeichnis `/tmp`, das vergrößert werden kann, um ausreichend Speicherplatz für den Befehl **bosboot** zu reservieren. Informationen zum Definieren der Variablen finden Sie in der Beschreibung der Datei `image.data`.

Wenn das System die Installation des Sicherungsimage abschließt, rekonfiguriert das Installationsprogramm den ODM auf dem Zielsystem. Sollte das Zielsystem nicht exakt dieselbe Hardwarekonfiguration wie das Quellsystem haben, ändert das Programm möglicherweise Einheitenattribute in den folgenden Dateien auf dem Zielsystem:

- Alle Dateien in `/etc/objrepos`, die mit `Cu` beginnen.
- Alle Dateien im Verzeichnis `/dev`.

Zugehörige Informationen:

Installing system backups

Geplante Sicherungen implementieren:

Diese Prozedur beschreibt, wie Sie ein Script entwickeln und verwenden, um eine wöchentliche vollständige Sicherung und tägliche inkrementelle Sicherungen von Benutzerdateien durchzuführen.

- Wenn Sie dieses Script verwenden, muss die für die Sicherung geplante Datenmenge auf ein Band passen.
- Stellen Sie sicher, dass das Band in die Sicherungseinheit eingelegt ist, bevor das Script mit dem Befehl **cron** ausgeführt wird.
- Stellen Sie sicher, dass die Einheit angeschlossen und verfügbar ist, insbesondere dann, wenn Sie Scripts verwenden, die nachts ausgeführt werden. Verwenden Sie den Befehl **lsdev -C | pg**, um die Verfügbarkeit zu prüfen.
- Stellen Sie sicher, dass seit der letzten Reinigung der Sicherungseinheit nicht zu viel Zeit vergangen ist, um Fehler zu verhindern.
- Wenn Sie Dateisysteme sichern möchten und diese Dateisysteme im Gebrauch sind, hängen Sie sie zuerst ab, um eine Beschädigung des Dateisystems zu vermeiden.
- Überprüfen Sie das Dateisystem, bevor Sie die Sicherung erstellen. Verwenden Sie die Prozedur „Dateisystem überprüfen“ auf Seite 470, oder führen Sie den Befehl **fsck** aus.

Das in dieser Prozedur verwendete Script ist nur als Modell gedacht und muss sorgfältig an die Anforderungen des jeweiligen Standorts angepasst werden.

Zugehörige Konzepte:

„Sicherungsstrategie“ auf Seite 26

Es gibt zwei Methoden für die Sicherung großer Datenmengen.

Dateisysteme mit dem Befehl cron sichern:

Diese Prozedur beschreibt, wie ein **crontab**-Script geschrieben wird, das Sie zur Ausführung an den Befehl **cron** übergeben können.

Das Script sichert zwei Benutzerdateisysteme, /home/plan und /home/run, jeweils in den Nächten von Montag bis Samstag. Beide Dateisysteme werden auf einem Band gesichert, und jeden Morgen wird ein neues Band für die folgende Nacht eingelegt. Die Sicherungen, die Montagnacht erstellt werden, sind vollständige Archive (Stufe 0). Die Sicherung von Dienstag bis Samstag sind inkrementelle Sicherungen.

1. Der erste Schritt beim Erstellen des **crontab**-Script ist das Absetzen des Befehls **crontab -e**. Dieser Befehl öffnet eine leere Datei, in der Sie Einträge einfügen können, die zur nächtlichen Ausführung an das **cron**-Script übergeben werden (der Standardeditor ist **vi**). Geben Sie Folgendes ein:

```
crontab -e
```

2. Das folgende Beispiel zeigt die sechs **crontab**-Felder. Feld 1 steht für die Minute, Feld 2 für die Stunde (24-Stunden-Format), Feld 3 für den Tag im Monat und Feld 4 für den Monat im Jahr. Die Felder 3 und 4 enthalten einen Stern (*), der angibt, dass das Script jeden Monat an dem im Feld **day/wk** angegebenen Tag ausgeführt wird. Feld 5 steht für den Tag der Woche. In diesem Feld kann auch ein Bereich von Tagen angegeben werden, z. B. 1-6. Feld 6 enthält den auszuführenden Shellbefehl.

```
min hr day/mo mo/yr day/wk      shell command
```

```
0 2 * * 1      backup -0 -uf /dev/rmt0.1 /home/plan
```

Die gezeigte Befehlszeile setzt voraus, dass Personal am Standort verfügbar ist, das gegebenenfalls auf Systemanfragen antworten kann. Das Flag **-0** (null) für den Befehl **backup** steht für Stufe 0 bzw. vollständige Sicherung. Das Flag **-u** aktualisiert den Sicherungssatz in der Datei /etc/dumpdates, und das Flag **f** gibt den Einheitennamen an, z. B. eine unformatierte Magnetbandeinheit 0.1 wie im obigen Beispiel.

3. Geben Sie eine Zeile wie die in Schritt 2 für jedes Dateisystem ein, das an einem bestimmten Tag gesichert werden soll. Das folgende Beispiel zeigt ein vollständiges Script, das an sechs Tagen zwei Dateisysteme sichert:

```
0 2 * * 1 backup -0 -uf/dev/rmt0.1 /home/plan
0 3 * * 1 backup -0 -uf/dev/rmt0.1 /home/run
0 2 * * 2 backup -1 -uf/dev/rmt0.1 /home/plan
0 3 * * 2 backup -1 -uf/dev/rmt0.1 /home/run
0 2 * * 3 backup -2 -uf/dev/rmt0.1 /home/plan
0 3 * * 3 backup -2 -uf/dev/rmt0.1 /home/run
0 2 * * 4 backup -3 -uf/dev/rmt0.1 /home/plan
0 3 * * 4 backup -3 -uf/dev/rmt0.1 /home/run
0 2 * * 5 backup -4 -uf/dev/rmt0.1 /home/plan
0 3 * * 5 backup -4 -uf/dev/rmt0.1 /home/run
0 2 * * 6 backup -5 -uf/dev/rmt0.1 /home/plan
0 3 * * 6 backup -5 -uf/dev/rmt0.1 /home/run
```

4. Speichern Sie die erstellte Datei und beenden Sie den Editor. Das Betriebssystem übergibt die Datei **crontab** an das **cron**-Script.

Zugehörige Informationen:

rmt Special File

Dateien auf einem mit DMAPI verwalteten JFS2-Dateisystem sichern:

Es werden Optionen für die Befehle **tar** und **backbyinode** unterstützt, mit denen Sie die erweiterten Attribute sichern können.

Mit dem Befehl **backbyinode** für ein DMAPI-Dateisystem werden nur die Daten gesichert, die zu dem Zeitpunkt, zu dem der Befehl abgesetzt wird, im Dateisystem vorhanden sind. Der Befehl **backbyinode** überprüft bei seiner Arbeit den aktuellen Status der Metadaten. Dies kann bei DMAPI von Vorteil sein, weil damit der Status des verwalteten Dateisystems gesichert wird. Nicht verfügbare Daten werden jedoch nicht gesichert.

Wenn Sie alle Daten in einem DMAPI-Dateisystem sichern möchten, müssen Sie einen Befehl verwenden, der vollständige Dateien liest, z. B. den Befehl **tar**. Dies kann dazu führen, dass eine DMAPI-fähige Anwendung Daten für jede Datei, auf die der Befehl **tar** zugreift, wiederherstellt und Daten zwischen dem sekundären und tertiären Speicher hin- und her verschiebt, was Auswirkungen auf die Leistung haben kann.

Disketten formatieren (Befehle **format** und **fdformat**):

Mit den Befehlen **format** und **fdformat** können Sie Disketten in dem für den Parameter *Einheit* angegebenen Diskettenlaufwerk (standardmäßig die Einheit `/dev/rfd0`) formatieren.

Achtung: Beim Formatieren werden sämtliche auf der Diskette gespeicherten Daten zerstört.

Der Befehl **format** erkennt die folgenden Einheitentypen:

- 5,25-Zoll-Diskette (360 KB) mit niedriger Speicherdichte und 40x2 Spuren mit jeweils 9 Sektoren
- 5,25-Zoll-Diskette (1,2 MB) mit hoher Kapazität und 80x2 Spuren mit jeweils 15 Sektoren
- 3,5-Zoll-Diskette (720 KB) mit niedriger Speicherdichte und 80x2 Spuren mit jeweils 9 Sektoren
- 3,5-Zoll-Diskette (2,88 MB) mit hoher Kapazität und 80x2 Spuren mit jeweils 36 Sektoren

Die Sektorgröße ist für alle Diskettentypen 512 Bytes.

Mit dem Befehl **format** können Sie Disketten mit hoher Speicherdichte formatieren, sofern für die *Einheit* keine andere Dichte angegeben ist.

Mit dem Befehl **fdformat** können Sie eine Diskette für niedrige Speicherdichte formatieren, sofern das Flag **-h** nicht angegeben ist. Sie geben mit dem Befehl die zu formatierende *Einheit* an, z. B. `/dev/rfd0` für Laufwerk 0.

Die Befehle **format** und **fdformat** fordern Sie vor dem Formatieren einer Diskette zur Bestätigung der Operation auf. Dies gibt Ihnen die Möglichkeit, bei Bedarf die Operation ordnungsgemäß zu beenden.

Beispiele:

- Geben Sie Folgendes ein, um eine Diskette in der Einheit `/dev/rfd0` zu formatieren:
`format -d /dev/rfd0`
- Geben Sie Folgendes ein, um eine Diskette ohne Überprüfung auf fehlerhafte Spuren zu formatieren:
`format -f`
- Geben Sie Folgendes ein, um eine 360-KB-Diskette in einem 5,25-Zoll-Diskettenlaufwerk, das für 1,2 MB ausgelegt ist, in der Einheit `/dev/rfd1` zu formatieren:
`format -l -d /dev/rfd1`
- Geben Sie Folgendes ein, um die Formatierung für hohe Speicherdichte bei Verwendung des Befehls **fdformat** zu erzwingen:
`fdformat -h`

Die vollständige Syntax des Befehls **format** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

Integrität eines Dateisystems überprüfen (Befehl **fsck**):

Mit dem Befehl **fsck** können Sie die Konsistenz von Dateisystemen überprüfen und mögliche Korrekturen interaktiv ausführen.

Dieser Befehl muss im Rahmen der Systeminitialisierung für jedes Dateisystem ausgeführt werden. Sie müssen die Einheitendatei, in der sich das Dateisystem befindet (z. B. die Einheit `/dev/hd0`), lesen kön-

nen. Das Dateisystem ist normalerweise konsistent, und der Befehl **fsck** zeigt nur die Anzahl der Dateien sowie die benutzten und freien Blöcke im Dateisystem an. Ist das Dateisystem inkonsistent, zeigt der Befehl **fsck** Informationen über die festgestellten Inkonsistenzen an und fordert Sie auf, die Berechtigung zur Korrektur zu erteilen. Bei der Ausführung des Befehls **fsck** werden Korrekturen äußerst vorsichtig vorgenommen und Aktionen vermieden, die den Verlust gültiger Daten nach sich ziehen könnten. In bestimmten Fällen empfiehlt der Befehl **fsck** jedoch, die beschädigte Datei zu löschen.

Achtung: Führen Sie den Befehl **fsck** nach jeder Systemstörung aus. Fehlerberichtigungen können zu einem Datenverlust führen. Standardmäßig wird bei jeder Konsistenzkorrektur darauf gewartet, dass der Bediener ja oder nein eingibt. Falls Sie keine Schreibberechtigung (*write*) für die betroffene Datei haben, verwendet der Befehl **fsck** standardmäßig die Antwort nein.

Beispiele:

- Geben Sie Folgendes ein, um alle Standarddateisysteme zu überprüfen:

```
fsck
```

Wenn Sie den Befehl **fsck** in dieser Form eingeben, wird Ihre Zustimmung angefordert, bevor Änderungen am Dateisystem vorgenommen werden.

- Geben Sie Folgendes ein, damit kleinere Probleme in den Standarddateisystemen automatisch korrigiert werden:

```
fsck -p
```

- Geben Sie Folgendes ein, um das Dateisystem `/dev/hd1` zu prüfen:

```
fsck /dev/hd1
```

Dieser Befehl überprüft das abgehängte Dateisystem auf der Einheit `/dev/hd1`.

Anmerkung: Der Befehl **fsck** nimmt keine Korrekturen an einem angehängten Dateisystem vor.

Die vollständige Syntax des Befehls **fsck** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

Daten auf oder von Disketten kopieren (Befehl **fcopy**):

Verwenden Sie den Befehl **fcopy**, um eine Diskette (die als `/dev/rfd0` geöffnet wurde) in eine Datei mit dem Namen `floppy` im aktuellen Verzeichnis zu kopieren.

Die Nachricht `Change floppy, hit return when done` erscheint, wenn die Diskette gewechselt werden muss. Anschließend kopiert der Befehl **fcopy** die Datei `floppy` auf die Diskette.

Beispiele:

- Geben Sie Folgendes ein, um `/dev/rfd1` in die Datei `floppy` im aktuellen Verzeichnis zu kopieren:

```
fcopy -f /dev/rfd1 -r
```

- Geben Sie Folgendes ein, um die ersten 100 Spuren der Diskette zu kopieren:

```
fcopy -f /dev/rfd1 -t 100
```

Die vollständige Syntax des Befehls **fcopy** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

Dateien auf Band oder Platte kopieren (Befehl **cpio -o**):

Mit dem Befehl **cpio -o** können Sie die Pfadnamen von Dateien aus der Standardeingabe lesen und diese Dateien zusammen mit den Pfadnamen und Statusinformationen in die Standardausgabe kopieren.

Pfadnamen haben eine maximal zulässige Länge von 128 Zeichen. Geben Sie mit dem Befehl **cpio** keine Pfadnamen an, die viele eindeutig verknüpfte Dateien enthalten, weil möglicherweise nicht genug Hauptspeicher verfügbar ist, um alle Dateipfade zu verfolgen. Dabei könnten Verbindungsdateien verloren gehen.

Beispiele:

- Geben Sie Folgendes ein, um die Dateien im aktuellen Verzeichnis, deren Name auf `.c` endet, auf eine Diskette zu kopieren:

```
ls *.c | cpio -ov >/dev/rfd0
```

Mit dem Flag `-v` werden die Namen der einzelnen Dateien angezeigt.

- Geben Sie Folgendes ein, um das aktuelle Verzeichnis einschließlich aller Unterverzeichnisse auf eine Diskette zu kopieren:

```
find . -print | cpio -ov >/dev/rfd0
```

Dieser Befehl sichert die Verzeichnisstruktur ab dem aktuellen Verzeichnis (`.`), einschließlich aller Unterverzeichnisse und Dateien.

- Wenn Sie eine kürzere Befehlszeichenfolge verwenden möchten, geben Sie Folgendes ein:

```
find . -cpio /dev/rfd0 -print
```

Die Angabe `-print` bewirkt, dass beim Kopieren der Dateien der jeweilige Name angezeigt wird.

Die vollständige Syntax des Befehls **cpio** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

Dateien von Band oder Platte kopieren (Befehl **cpio -i**):

Mit dem Befehl **cpio -i** können Sie eine Archivierungsdatei, die mit dem Befehl **cpio -o** erstellt wurde, aus der Standardeingabe lesen und aus dieser Datei die Dateien mit den Namen kopieren, die einem von Ihnen angegebenen *Muster* entsprechen.

Diese Dateien werden in die aktuelle Verzeichnisstruktur kopiert. Wenn Sie die im Abschnitt zum Befehl **ksh** in der Veröffentlichung "Commands Reference" beschriebene Dateinamennotation verwenden, können Sie mehrere *Muster* angeben. Standardmäßig wird als *Muster* ein Stern (*) verwendet, der alle Dateien im aktuellen Verzeichnis auswählt. In einem Ausdruck wie `[a-z]` bedeutet das Minuszeichen (-) entsprechend der aktuellen Sortierfolge *bis*.

Anmerkung: Die Muster `*.c` und `*.o` müssen in Anführungszeichen eingeschlossen werden, um zu verhindern, dass die Shell den Stern (*) als Platzhalterzeichen interpretiert. Dies ist ein Sonderfall, bei dem der Befehl **cpio** die Platzhalterzeichen selbst entschlüsselt.

Beispiele:

- Geben Sie Folgendes ein, um die Dateien aufzulisten, die mit dem Befehl **cpio** auf Diskette gesichert wurden:

```
cpio -itv </dev/rfd0
```

Dieser Befehl zeigt das Inhaltsverzeichnis der zuvor in der Datei `/dev/rfd0` gesicherten Daten im Format des Befehls **cpio** an. Die Liste gleicht der umfangreichen Verzeichnisliste, die mit dem Befehl **ls -l** erzeugt wird.

- Mit den Flags `-it` werden ausschließlich die Dateipfadnamen aufgelistet.
- Geben Sie Folgendes ein, um die zuvor mit dem Befehl **cpio** gesicherten Dateien von der Diskette zu kopieren:

```
cpio -idmv </dev/rfd0
```

Dieser Befehl kopiert die zuvor mit dem Befehl **cpio** in der Datei /dev/rfd0 gespeicherten Dateien in das Dateisystem zurück (geben Sie das Flag **-i** an). Durch Angabe des Flags **-d** werden mit dem Befehl **cpio** die entsprechenden Verzeichnisse beim Sichern einer Verzeichnisstruktur erstellt. Das Flag **-m** bewirkt, dass beim Speichern der Dateien das letzte Änderungsdatum übernommen wird. Mit dem Flag **-v** wird der Befehl **cpio** angewiesen, den Namen der Datei anzuzeigen, die gerade kopiert wird.

- Geben Sie Folgendes ein, um die ausgewählten Dateien von der Diskette zu kopieren:

```
cpio -i "*.c" "*.o" </dev/rfd0
```

Dieser Befehl kopiert die Dateien, die mit **.c** oder **.o** enden, von der Diskette.

Die vollständige Syntax des Befehls **cpio** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

Dateien auf oder von Bändern kopieren (Befehl tcopy):

Mit dem Befehl **tcopy** können Sie Magnetbänder kopieren.

Geben Sie Folgendes ein, um Daten von einem Datenstromband auf ein Band mit 9 Spuren zu kopieren:

```
tcopy /dev/rmt0 /dev/rmt8
```

Die vollständige Syntax des Befehls **tcopy** ist in der Veröffentlichung *Commands Reference, Volume 5* beschrieben.

Integrität eines Bandes überprüfen (Befehl tapechk):

Mit dem Befehl **tapechk** können Sie rudimentäre Konsistenzprüfungen für eine angeschlossene Datenstrombandeinheit durchführen.

Manche Fehlfunktionen der Hardware lassen sich einfach durch Lesen eines Bandes feststellen. Mit dem Befehl **tapechk** können Bänder auf Dateiebene gelesen werden.

Geben Sie Folgendes ein, um die ersten drei Dateien auf einer Datenstrombandeinheit zu überprüfen:

```
tapechk 3
```

Die vollständige Syntax des Befehls **tapechk** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

Dateien archivieren (Befehl tar):

Die Archivierungsmethode wird verwendet, um eine Kopie einer oder mehrerer Dateien oder einer gesamten Datenbank zu erstellen, die für spätere Referenz, zu Protokollierungszwecken oder für die Wiederherstellung gespeichert wird, falls die ursprünglichen Daten beschädigt werden oder verloren gehen.

Gewöhnlich werden Archive verwendet, wenn die entsprechenden Daten vom System entfernt werden.

Mit dem Befehl **tar** können Sie Dateien in einen Archivierungsspeicher schreiben und aus einem Archivierungsspeicher abrufen. Der Befehl **tar** sucht auf der Standardeinheit (gewöhnlich einem Band) nach Archiven, sofern Sie keine andere Einheit angeben.

Beim Schreiben in ein Archiv verwendet der Befehl **tar** eine temporäre Datei (die Datei /tmp/tar*) und verwaltet im Speicher eine Tabelle der Dateien, die mehrere Verbindungen besitzen. Kann die temporäre Datei mit dem Befehl **tar** nicht erstellt werden oder ist nicht genug Speicher für die Verbindungstabellen verfügbar, wird eine Fehlernachricht angezeigt.

Beispiele:

- Sie möchten die Dateien `Datei1` und `Datei2` in ein neues Archiv auf dem Standardbandlaufwerk schreiben. Geben Sie Folgendes ein:

```
tar -c Datei1 Datei2
```
- Sie möchten alle Dateien aus der Archivierungsdatei auf der Bandoeinheit `/dev/rmt2` in das Verzeichnis `/tmp` extrahieren und die Uhrzeit der Extraktion als Änderungszeit verwenden. Geben Sie Folgendes ein:

```
tar -xm -f/dev/rmt2 /tmp
```
- Sie möchten die Namen der Dateien in der Plattenarchivierungsdatei `out.tar` vom aktuellen Verzeichnis aus anzeigen. Geben Sie Folgendes ein:

```
tar -vtf out.tar
```

Weitere Informationen und eine Beschreibung der genauen Syntax finden Sie in den Erläuterungen zum Befehl `tar` in der Veröffentlichung *Commands Reference, Volume 5*.

Dateisicherung

Mit dem Befehl `backup` oder dem Befehl `smit` können Sie Kopien Ihrer Dateien auf einem Sicherungsdatenträger, wie beispielsweise einem Magnetband oder einer Diskette, erstellen.

Achtung: Wenn Sie versuchen, ein angehängtes Dateisystem zu sichern, wird eine Nachricht angezeigt. Der Befehl `backup` wird zwar weiter ausgeführt, aber es können Inkonsistenzen im Dateisystem auftreten. Diese Warnung gilt nicht für das Stammdateisystem (`/`).

Die Kopien, die Sie mit dem Befehl `backup` oder `smit` erstellen, haben eines der folgenden Sicherungsformate:

- Mit dem Flag `-i` können Sie bestimmte angegebene Dateien sichern.
- Mit den Parametern `-Stufe` und `Dateisystem` können Sie ein vollständiges Dateisystem nach I-Node-Nummer sichern.

Anmerkung:

- Wenn eine Datei während der Systemdatensicherung geändert wird, besteht potenziell immer das Risiko einer Datenbeschädigung. Stellen Sie deshalb sicher, dass während der Durchführung der Systemdatensicherung so wenig wie möglich Systemaktivitäten stattfinden.
- Wenn Sie die Sicherung auf ein 8-mm-Band schreiben und die Einheitenblockgröße auf 0 (null) eingestellt ist, können Daten nicht direkt vom Band zurückgeschrieben werden. Sollten Sie die Sicherungen mit der Einstellung 0 durchgeführt haben, können Sie die Sicherungsdaten mit speziellen Prozeduren, die im Abschnitt zum Befehl `restore` beschrieben werden, wiederherstellen.

Achtung: Stellen Sie sicher, dass die angegebenen Flags für den Sicherungsdatenträger geeignet sind.

Dateien mit dem Befehl `backup` sichern:

Mit dem Befehl `backup` können Sie Kopien Ihrer Dateien auf einem Sicherungsdatenträger erstellen.

Geben Sie beispielsweise Folgendes ein, um ausgewählte Dateien im Verzeichnis `$HOME` nach Namen zu sichern:

```
find $HOME -print | backup -i -v
```

Mit dem Flag `-i` wird das System angewiesen, die Namen der zu sichernden Dateien aus der Standardeingabe zu lesen. Der Befehl `find` generiert eine Liste der Dateien im Verzeichnis des Benutzers. Diese Liste wird über eine Pipe als Standardeingabe an den Befehl `backup` geleitet. Mit dem Flag `-v` wird nach jeder kopierten Datei ein Fortschrittsbericht angezeigt. Die Dateien werden auf der Standardsicherungseinheit für das lokale System gesichert.

Beispiele:

- Geben Sie Folgendes ein, um das Stammdateisystem zu sichern:

```
backup -0 -u /
```

Mit der Angabe der Sicherungsstufe 0 und eines Schrägstrichs (/) wird das System angewiesen, das Stammdateisystem (/) zu sichern. Das Dateisystem wird in der Datei /dev/rfd0 gesichert. Mit dem Flag **-u** wird das System angewiesen, die aktuelle Sicherheitsstufe in der Datei /etc/dumpdates zu aktualisieren.

- Geben Sie Folgendes ein, wenn Sie alle Dateien im Dateisystem / (Stammverzeichnis) sichern möchten, die seit der letzten Sicherung der Stufe 0 geändert wurden.

```
backup -1 -u /
```

Die vollständige Syntax des Befehls **backup** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

Dateien mit dem Befehl **smit** sichern:

Mit dem Befehl **smit** können Sie den Befehl **backup** ausführen, der Kopien Ihrer Dateien auf einem Sicherungsdatenträger erstellt.

1. Geben Sie an der Eingabeaufforderung Folgendes ein:

```
smit backup
```

2. Geben Sie im Feld **Vollständiger Pfadname des Verzeichnisses** den Pfadnamen des Verzeichnisses ein, an das das Dateisystem normalerweise angehängt ist:

```
/home/bill
```

3. Geben Sie im Feld **Sicherungseinheit** oder **Datei** den Namen der Ausgabeinheit ein. Im folgenden Beispiel wird ein unformatiertes Magnetband als Ausgabeinheit verwendet:

```
/dev/rmt0
```

4. Mit der Tabulatortaste können Sie die Einstellung des Feldes **Bericht über jede Phase der Sicherung** ändern, wenn Fehlernachrichten am Bildschirm ausgegeben werden sollen.
5. Verwenden Sie in einer Systemverwaltungsumgebung den Standardwert für das Feld **Maximale Anzahl der auf den Sicherungsdatenträger zu schreibenden Blöcke**, weil dieses Feld für Sicherungen auf Band nicht anwendbar ist.
6. Drücken Sie die Eingabetaste, um das angegebene Verzeichnis bzw. Dateisystem zu sichern.
7. Führen Sie den Befehl **restore -t** aus. Wird von diesem Befehl eine Fehlernachricht generiert, muss die gesamte Sicherung wiederholt werden.

Systemabschluss durchführen

Der Befehl **shutdown** ist die sicherste und gründlichste Methode zum Anhalten des Betriebssystems.

Sie können Ihr System in den folgenden Fällen herunterfahren:

- Nach der Installation neuer Software oder nach dem Ändern der Konfiguration vorhandener Software
- Beim Vorliegen eines Hardwarefehlers
- Bei einer irreversiblen Blockierung des Systems
- Bei einer verminderten Systemleistung
- Bei einer möglichen Beschädigung des Dateisystems

Wenn Sie die entsprechenden Flags angeben, benachrichtigt dieser Befehl Benutzer darüber, dass das System heruntergefahren wird, beendet alle vorhandene Prozesse, hängt Dateisysteme ab und hält das System an. Weitere Informationen finden Sie in der Beschreibung des Befehls **shutdown**.

Die folgenden Informationen enthalten Details zu bestimmten Systemabschlussituationen:

System ohne Warmstart herunterfahren

Es gibt zwei Methoden, ein System ohne Warmstart herunterzufahren.

Ihnen stehen zwei Methoden zur Verfügung, das System ohne Warmstart herunterzufahren: der SMIT-Direktaufruf und der Befehl **shutdown**.

Voraussetzungen

Sie müssen Rootberechtigung besitzen, um das System herunterzufahren.

Gehen Sie wie folgt vor, um das System mit SMIT herunterzufahren:

1. Melden Sie sich als Root an.
2. Geben Sie an der Eingabeaufforderung Folgendes ein:
smit shutdown

Gehen Sie wie folgt vor, um das System mit dem Befehl **shutdown** herunterzufahren:

1. Melden Sie sich als Root an.
2. Geben Sie an der Eingabeaufforderung Folgendes ein:
shutdown

System in den Einzelbenutzermodus herunterfahren

Manchmal müssen Sie das System herunterfahren und im Einzelbenutzermodus ausführen, um Softwareverwaltungs- und Diagnosetasks auszuführen.

1. Geben Sie `cd /` ein, um in das Stammverzeichnis zu wechseln. Sie müssen im Stammverzeichnis sein, um das System in den Einzelbenutzermodus herunterzufahren und sicherzustellen, dass die Dateisysteme ordnungsgemäß abgehängt werden.
2. Geben Sie `shutdown -m` ein. Das System wird in den Einzelbenutzermodus heruntergefahren.

Es erscheint eine Eingabeaufforderung, und Sie können Verwaltungsaktivitäten ausführen.

System in einem Notfall herunterfahren

Verwenden Sie den Befehl **shutdown**, wenn Sie das System schnell stoppen müssen. In diesem Fall werden die anderen Benutzer nicht benachrichtigt.

Mit dem Befehl **shutdown** können Sie das System in einem Notfall herunterfahren.

Geben Sie `shutdown -F` ein. Das Flag `-F` weist den Befehl **shutdown** an, das Senden von Nachrichten an andere Benutzer zu umgehen und das System so schnell wie möglich herunterzufahren.

Systemumgebung

Die Systemumgebung setzt sich primär aus der Gruppe von Variablen zusammen, die bestimmte Aspekte der Prozessausführung definieren oder steuern.

Sie werden jedes Mal, wenn eine Shell gestartet wird, gesetzt bzw. zurückgesetzt. Aus der Perspektive der Systemverwaltung ist es wichtig sicherzustellen, dass für den Benutzer bei der Anmeldung die richtigen Werte konfiguriert werden. Die meisten dieser Variablen werden während der Systeminitialisierung gesetzt. Ihre Definitionen werden aus der Datei `/etc/profile` gelesen oder standardmäßig gesetzt.

Profile

Die Shell verwendet beim Anmelden am Betriebssystem zwei Arten von Profildateien.

Die Shell wertet die in den Dateien enthaltenen Befehle aus und führt anschließend die Befehle zum Einrichten Ihrer Systemumgebung aus. Die Dateien haben ähnliche Funktionen. Allerdings steuert die Datei `/etc/profile` Profilvariablen für alle Benutzer auf einem System, während Sie mit der Datei `.profile` Ihre eigene Umgebung anpassen können.

Die folgenden Profil- und Systemumgebungsdaten werden bereitgestellt:

- Datei /etc/profile
- Datei .profile
- Konfiguration der Systemumgebungsvariablen
- Nachricht des Tages ändern
- „Services für die Einstellung von Uhrzeit und Datum“.

Datei /etc/profile

Die erste Datei, die das Betriebssystem zur Anmeldezeit verwendet, ist die Datei /etc/profile. Diese Datei steuert systemweite Standardvariablen, z. B.:

- Exportvariablen
- Dateierstellungsmaske (umask)
- Terminaltypen
- Mail-Nachrichten, die anzeigen, dass neue Mail empfangen wurde

Der Systemadministrator konfiguriert die Datei profile für alle Benutzer auf dem System. Anschließend der Systemadministrator ist berechtigt, die Datei zu ändern.

Datei .profile

Die zweite Datei, die das Betriebssystem zur Anmeldezeit verwendet, ist die Datei .profile. Die Datei .profile befindet sich in Ihrem Ausgangsverzeichnis (\$HOME) und ermöglicht Ihnen, Ihre Arbeitsumgebung individuell anzupassen. Die Datei .profile überschreibt außerdem Befehle und Variablen, die in der Datei /etc/profile definiert sind. Da die Datei .profile verdeckt gespeichert ist, müssen Sie zum Auflisten der Datei den Befehl `ls -a` verwenden. Sie können mit der Datei .profile die folgenden Standardeinstellungen steuern:

- Zu öffnende Shells
- Eingabeaufforderung
- Umgebungsvariablen (z. B. Variablen für den Suchpfad)
- Tastatursignale

Das folgende Beispiel zeigt eine typische Datei .profile:

```
PATH=/usr/bin:/etc:/home/bin1:/usr/lpp/tps4.0/user:/home/gsc/bin::
epath=/home/gsc/e3:
export PATH epath
csh
```

In dieser Beispieldatei werden zwei Pfade definiert (PATH und epath) und exportiert. Anschließend wird eine C-Shell (csh) geöffnet.

Zum Bestimmen der Variablen für die Anmeldeshell können Sie auch die Datei .profile (oder, sofern diese Datei nicht vorhanden ist, die Datei /etc/profile) verwenden. Andere Shellumgebungen können ebenfalls angepasst werden. Verwenden Sie die Dateien .chsrc und .kshrc beispielsweise, um eine C-Shell bzw. eine Korn-Shell anzupassen, wenn jeweils ein Typ jeder Shell gestartet wird.

Services für die Einstellung von Uhrzeit und Datum

Die Zeitfunktionen greifen auf das aktuelle Systemdatum und die aktuelle Systemzeit zu und formatieren diese erneut.

Sie müssen kein spezielles Flag für den Compiler angeben, damit die Zeitfunktionen verwendet werden. Fügen Sie die Header-Datei für diese Funktionen in das Programm ein. Verwenden Sie zum Einfügen einer Header-Datei die folgende Anweisung:

```
#include <time.h>
```

Die folgenden Zeitservices sind verfügbar:

Eintrag	Beschreibung
<code>adjtime</code>	Korrigiert die Zeit, um eine Synchronisation der Systemuhr durchzuführen.
<code>ctime, localtime, gmtime, mktime, difftime, asctime, tzset getinterval, incinterval, absinterval, resinc, resabs, alarm, ualarm, getitimer, setitimer</code>	Konvertieren Datum und Uhrzeit in eine Zeichenfolgedarstellung. Bearbeiten die Verfallszeit von Intervallzeitgebern.
<code>gettimer, settimer, restimer, stime, time</code>	Rufen den aktuellen Wert für den angegebenen systemweiten Zeitgeber ab oder setzen ihn.
<code>gettimerid</code>	Ordnet einen prozesseigenen Intervallzeitgeber zu.
<code>gettimeofday, settimeofday, ftime</code>	Rufen Datum und Uhrzeit ab und setzen sie.
<code>nsleep, usleep, sleep</code>	Setzen die Ausführung eines aktuellen Prozesses aus.
<code>realtimerid</code>	Gibt einen zuvor zugeordneten Intervallzeitgeber frei.

Für den 64-Bit-Modus erforderliche Dateien und Hardware

Der Kernel wird im 64-Bit-Modus ausgeführt. Dies ermöglicht einen schnellen Zugriff auf große Datenvolumen und eine effiziente Handhabung von 64-Bit-Datentypen.

Die 64-Bit-Laufzeitdateigruppe für das Basisbetriebssystem ist `bos.64bit`. Wenn Sie `bos.64bit` installieren, wird auch die Datei `/etc/methods/cfg64` installiert. Die Datei `/etc/methods/cfg64` ist ein Befehl, der die 64-Bit-Laufzeitumgebung aktiviert. Dieser Befehl wird vom Script `rc.boot` in Phase 3 des Bootprozesses aufgerufen.

Beginnend mit AIX 6.1 wird der 32-Bit-Kernel nicht weiter unterstützt. Mit der Installation des Basisbetriebssystems AIX 6.1 wird der 64-Bit-Modus aktiviert.

Anmerkung: Die Hardware muss 64-Bit-fähig sein, damit AIX 6.1 ausgeführt werden kann. Die folgenden RS/6000-Modelle verwenden 604e-Prozessoren und sind nicht 64-Bit-fähig:

- 7025 F50 Series
- 7026 H50 Series
- 9076 H50 Series
- 7043 150 Series
- 7046 B50 Series

Führen Sie den folgenden Befehl aus, um die Funktionalität des Prozessors zu ermitteln:

```
/usr/sbin/prtconf -c
```

Der Befehl `prtconf` gibt abhängig von der Funktionalität des Prozessors entweder 32 oder 64 zurück. Falls Ihr System den Befehl `prtconf` nicht unterstützt, verwenden Sie den Befehl `bootinfo` mit dem Flag `-y`.

Für den 64-Bit-Modus erforderliche Hardware

Für die Ausführung von 64-Bit-Anwendungen benötigen Sie 64-Bit-Hardware.

Gehen Sie wie folgt vor, um festzustellen, ob Ihr System eine 32-Bit- oder 64-Bit-Hardwarearchitektur hat:

1. Melden Sie sich als Root an.
2. Geben Sie in der Befehlszeile `bootinfo -y` ein.

Dieser Befehl gibt entweder **32** (für eine 32-Bit-Hardwarearchitektur) oder **64** (für eine 64-Bit-Hardwarearchitektur) aus. Mit dem Befehl `lsattr -El proc0` können Sie in jeder Version von AIX den Typ des Prozessors für Ihren Server anzeigen.

32-Bit- und 64-Bit-Leistung im Vergleich

In den meisten Fällen ist die Ausführung von 32-Bit-Anwendungen auf 64-Bit-Hardware kein Problem, da auf 64-Bit-Hardware sowohl 64-Bit- als auch 32-Bit-Software ausgeführt werden kann. Auf 32-Bit-Hardware kann jedoch keine 64-Bit-Software ausgeführt werden.

Um festzustellen, ob Leistungsprobleme bei Anwendungen zu verzeichnen sind, die auf dem System ausgeführt werden, lesen Sie in den Benutzerhandbüchern zu diesen Anwendungen nach, welche Ausführungsumgebung empfohlen wird.

Dynamische Prozessorfreigabe

AIX kann einen fehlerhaften Prozessor erkennen und automatisch stoppen.

Seit dem Maschinentyp IBM 7044 Modell 270 ist die Hardware aller Systeme mit zwei Prozessoren in der Lage, behebbare Fehler zu erkennen, die von der Firmware erfasst werden. Diese Fehler sind nicht fatal und können, solange sie nur selten auftreten, unbesorgt ignoriert werden. Wenn sich jedoch ein Fehlermuster in einem bestimmten Prozessor zu entwickeln scheint, kann dieses Muster darauf hinweisen, dass in naher Zukunft ein fataler Fehler in der Komponente auftritt. Diese Vorhersage wird von der Firmware basierend auf der Analyse von Fehlerraten und -schwellenwerten erstellt.

AIX implementiert auf diesen Systemen eine fortlaufende Hardwareüberwachung und fragt die Firmware regelmäßig auf Hardwarefehler ab. Wenn die Anzahl der Prozessorfehler einen Schwellenwert erreicht und die Firmware erkennt, dass eine hohe Wahrscheinlichkeit besteht, dass diese Systemkomponente ausfällt, gibt die Firmware einen Fehlerbericht zurück. In jedem Fall wird der Fehler im Systemfehlerprotokoll aufgezeichnet. Außerdem versucht AIX auf Mehrprozessorsystemen je nach Fehlertyp, den nicht mehr vertrauenswürdigen Prozessor zu stoppen und freizugeben. Diese Funktion hat den Namen *Dynamische Prozessorfreigabe*.

Gleichzeitig wird der Prozessor von der Firmware so lange für eine persistente Freigabe bei nachfolgenden Bootvorgängen markiert, bis das Wartungspersonal den Prozessor austauscht.

Auswirkungen der Prozessorfreigabe auf Anwendungen:

Die Prozessorfreigabe ist für die große Mehrheit der Anwendungen, einschließlich Treiber und Kernelerweiterungen transparent. Mit den veröffentlichten Schnittstellen können Sie jedoch feststellen, ob eine Anwendung oder Kernelerweiterung auf einer Multiprozessormaschine ausgeführt wird und wie viele Prozessoren vorhanden sind, und Threads an bestimmte Prozessoren binden.

Die Schnittstelle `bindprocessor` für das Binden von Prozessen oder Threads an Prozessoren verwendet CPU-Nummern für die Bindung. Der gültige Wertebereich für die CPU-Nummern ist $[0..N-1]$, wobei N für die Gesamtanzahl der CPUs steht. Damit die Ausführung von Anwendungen oder Kernelerweiterungen, die von einer durchgängigen CPU-Nummerierung ausgehen, nicht unterbrochen wird, spiegelt AIX den Anwendungen vor, dass es sich um die "letzte" CPU (mit der höchsten Nummer) handelt, die freigegeben werden soll. Auf einem 8-Wege-SMP-System sind die CPU-Nummern beispielsweise $[0..7]$. Wenn ein Prozessor freigegeben wird, ist die Gesamtanzahl der verfügbaren CPUs 7, die von $[0..6]$ nummeriert sind. Extern scheint es so, dass CPU 7 nicht mehr vorhanden ist, unabhängig davon, welcher physische Prozessor ausgefallen ist.

Anmerkung: Im verbleibenden Teil dieser Beschreibung werden der Begriff *CPU* für die logische Entität und der Begriff *Prozessor* für die physische Entität verwendet.

Potenziell können Anwendungen oder Kernelerweiterungen, die gebundene Prozesse oder Threads sind, unterbrochen werden, wenn AIX ihre gebundenen Threads beendet oder explizit in eine andere CPU verschiebt, falls einer der Prozessoren freigegeben werden muss. Die Funktion "Dynamische Prozessorfreigabe" stellt Programmierschnittstellen bereit, so dass Anwendungen und Kernelerweiterungen darüber benachrichtigt werden können, wenn eine Prozessorfreigabe bevorsteht. Wenn diese Anwendungen und Kernelerweiterungen eine Benachrichtigung erhalten, sind sie dafür verantwortlich, ihre gebundenen Threads und zugeordneten Ressourcen (z. B. Anforderungsblöcke für Zeitgeber) von der letzten gebundenen CPU-ID zu entfernen und sich an die neue CPU-Konfiguration anzupassen.

Wenn einige Threads nach der Benachrichtigung an die letzte CPU-ID gebunden bleiben, wird die Freigabe abgebrochen. Die abgebrochene Freigabe wird im Fehlerprotokoll aufgezeichnet, und AIX verwendet

weiterhin den fehlerhaften Prozessor. Wenn der Prozessor schließlich ausfällt, führt dies zu einem Ausfall des Gesamtsystems. Deshalb ist es wichtig, dass Anwendungen und Kernelerweiterungen über eine bevorstehende Prozessorfreigabe informiert werden und entsprechend handeln können.

Selbst wenn die Freigabe nicht durchgeführt werden kann, warnt die dynamische Prozessorfreigabe die Systemadministratoren im Vorfeld. Durch die Aufzeichnung des Fehlers im Fehlerprotokoll erhalten die Systemadministratoren die Möglichkeit, eine Wartungsoperation für das System zu planen, um die defekte Komponente auszutauschen, bevor ein globaler Systemausfall eintritt.

Prozessorfreigabeprozess:

AIX kann einen fehlerhaften Prozessor durch Freigabe stoppen.

Der typische Ereignisablauf bei der Prozessorfreigabe wird im Folgenden beschrieben:

1. Die Firmware erkennt, dass einer der Prozessoren einen Schwellenwert für behebbare Fehler erreicht hat.
2. Der Fehlerbericht der Firmware wird im Systemfehlerprotokoll aufgezeichnet. Wenn AIX auf einer Maschine ausgeführt wird, die die Prozessorfreigabe unterstützt, startet AIX den Freigabeprozess.
3. AIX benachrichtigt alle Prozesse und Threads, die keine Kernelprozesse bzw. Kernel-Threads sind, die an die letzte BIND-CPU gebunden sind.
4. AIX gewährt allen gebundenen Threads einen Zeitraum von bis zu zehn Minuten, um sich von der letzten BIND-CPU zu lösen. Wenn die Threads gebunden bleiben, bricht AIX die Freigabe ab.
5. Wenn alle Prozesse und Threads vom fehlerhaften Prozess gelöst sind, werden die zuvor registrierten High Availability Event Handlers (HAEHs) aufgerufen. Ein HAEH kann einen Fehler zurückgeben, der die Freigabe abbricht.
6. Sofern der Freigabeprozess nicht abgebrochen wird, stoppt er den fehlerhaften Prozessor.

Sollte während der Freigabe ein Fehler auftreten, werden der Fehler und die Ursache protokolliert. Der Systemadministrator kann das Fehlerprotokoll untersuchen, (sofern möglich) Maßnahmen zur Fehlerbehebung ergreifen und die Freigabe erneut starten. Wenn die Freigabe beispielsweise abgebrochen wurde, weil eine Anwendung ihre gebundenen Threads nicht entfernt hat, kann der Systemadministrator die Anwendung stoppen und die Freigabe und anschließend die Anwendung erneut starten.

Dynamische Prozessorfreigabe aktivieren:

Wenn Ihre Maschine die Funktion "Dynamische Prozessorfreigabe" unterstützt, können Sie SMIT oder Systembefehle verwenden, um die Funktion zu **aktivieren** oder zu **inaktivieren**.

Die Funktion "Dynamische Prozessorfreigabe" wird standardmäßig während der Installation aktiviert, sofern die Maschine die korrekte Hardware und Firmware hat, die diese Funktion unterstützt.

Prozedur mit dem SMIT-Direktaufruf

1. Geben Sie an der Eingabeaufforderung den Befehl `smit system` ein, und drücken Sie anschließend die Eingabetaste. Hierfür benötigen Sie Rootberechtigung.
2. Wählen Sie im Fenster **Systemumgebung** die Option **Merkmale des Betriebssystems ändern/ anzeigen** aus.
3. Verwenden Sie die SMIT-Dialoge, um die Tasks auszuführen.

Wenn Sie zusätzliche Informationen für die Ausführung der Task benötigen, können Sie die Hilfetaste F1 in den SMIT-Dialogen auswählen.

Prozedur mit Befehlen

Wenn Sie Rootberechtigung besitzen, können Sie mit den folgenden Befehlen die Funktion "Dynamische Prozessorfreigabe" bearbeiten:

- Verwenden Sie den Befehl **chdev**, um die Merkmale der angegebenen Einheit zu ändern. Weitere Verwendungsinformationen finden Sie in der Beschreibung des Befehls **chdev** in der Veröffentlichung *Commands Reference, Volume 1*.
- Sollte die Prozessorfreigabe aus irgendeinem Grund scheitern, können Sie den Prozessor mit dem Befehl **ha_star** nach der Fehlerbehebung erneut starten. Weitere Verwendungsinformationen finden Sie in der Beschreibung des Befehls **ha_star** in der Veröffentlichung *Commands Reference, Volume 2*.
- Verwenden Sie den Befehl **errpt**, um einen Bericht über die protokollierten Fehler zu generieren. Weitere Verwendungsinformationen finden Sie in der Beschreibung des Befehls **errpt** in der Veröffentlichung *Commands Reference, Volume 2*.

Methoden für die Aktivierung und Inaktivierung der Prozessorfreigabe:

Die Funktion "Dynamische Prozessorfreigabe" kann aktiviert oder inaktiviert werden, indem der Wert des Attributs **cpuguard** des ODM-Objekts `sys0` geändert wird.

Die gültigen Werte für das Attribut sind `enable` und `disable`.

Die Standardeinstellung ist `enabled` (das Attribut **cpuguard** hat den Wert `enable`). Systemadministratoren, die die Funktion inaktivieren möchten, müssen hierfür die Systemmenüs, das SMIT-Menü **Systemumgebungen** oder den Befehl **chdev** verwenden. (In den früheren Versionen von AIX ist die Standardeinstellung `disabled`.)

Anmerkung: Wenn die Prozessorfreigabe inaktiviert ist, werden trotzdem Fehler protokolliert. Das Fehlerprotokoll enthält einen Fehler wie `CPU_FAILURE_PREDICTED`, der anzeigt, dass AIX über ein Problem mit der CPU benachrichtigt wurde.

Eine abgebrochene Prozessorfreigabe erneut starten:

Manchmal scheitert die Freigabe des Prozessors, weil eine Anwendung ihre gebundenen Threads nicht von der letzten logischen CPU entfernt.

Sobald dieses Problem behoben wird, entweder durch Aufheben der Bindung (sofern dies sicher ist) oder durch Stoppen der Anwendung, kann der Systemadministrator die Prozessorfreigabe mit dem Befehl **ha_star** erneut starten.

Der Befehl hat die folgende Syntax:

```
ha_star -C
```

`-C` steht für ein vorhersehbares CPU-Fehlerereignis.

Hinweise zum Prozessorstatus:

In Bezug auf Prozessorstatus sind verschiedene Dinge zu berücksichtigen.

Physische Prozessoren werden in der ODM-Datenbank durch Objekte mit dem Namen **procn** dargestellt, wobei *n* für eine Dezimalzahl steht, die die Nummer des physischen Prozessors angibt. Wie andere Einheiten, die in der ODM-Datenbank dargestellt werden, haben Prozessorobjekte einen Status, z. B. `Defined` oder `Available` (Definiert oder Verfügbar), und Attribute.

Der Status eines Objekts **proc** ist immer Available, sofern der entsprechende Prozessor vorhanden ist, unabhängig davon, ob er verwendbar ist oder nicht. Das Statusattribut (**state**) eines Objekts **proc** gibt an, ob der Prozessor verwendet wird, und wenn nicht, den Grund dafür. Die drei gültigen Werte für dieses Attribut sind im Folgenden beschrieben:

Eintrag	Beschreibung
enable	Der Prozessor wird verwendet.
disable	Der Prozessor wurde dynamisch freigegeben.
faulty	Der Prozessor wurde von der Firmware beim Start als fehlerhaft deklariert.

Wenn ein fehlerhafter Prozessor erfolgreich freigegeben wird, wechselt der Status von **enable** in **disable**. Unabhängig von AIX wird dieser Prozessor außerdem in der Firmware als fehlerhaft gekennzeichnet. Nach einem Warmstart ist der freigegebene Prozessor nicht verfügbar und erhält den Status **faulty**. Das ODM-Objekt **proc** ist jedoch weiterhin als verfügbar gekennzeichnet. Sie müssen die fehlerhafte CPU physisch von der Systemplatine entfernen oder (sofern möglich) die CPU-Platine entfernen, damit der Status des Objekts **proc** in Defined geändert wird.

Im folgenden Beispiel arbeitet der Prozessor **proc4** ordnungsgemäß und wird vom Betriebssystem verwendet, wie die folgende Ausgabe zeigt:

```
# lsattr -EH -l proc4
attribute value  description user_settable

state enable Processor state False
type PowerPC_RS64-III Processor type False
#
```

Wenn für den Prozessor **proc4** ein bevorstehender Ausfall gemeldet wird, wird er vom Betriebssystem freigegeben, wie das folgende Beispiel zeigt:

```
# lsattr -EH -l proc4
attribute value  description user_settable

state disable Processor state False
type PowerPC_RS64-III Processor type False
#
```

Beim nächsten Systemwiederanlauf wird der Prozessor **proc4** von der Firmware als fehlerhaft gemeldet, wie das folgende Beispiel zeigt:

```
# lsattr -EH -l proc4
attribute value  description user_settable

state faulty Processor state False
type PowerPC_RS64-III Processor type False
#
```

Der Prozessor **proc4** behält jedoch weiterhin den Status Available, wie das folgende Beispiel zeigt.

```
# lsdev -CH -l proc4
name status location description

proc4 Available 00-04 Processor
#
```

Fehlerprotokolleinträge zur Freigabe:

Bei der CPU-Freigabe können drei unterschiedliche Fehlerprotokollnachrichten ausgegeben werden.

Im Folgenden sehen Sie verschiedene Beispiele:

Kurzformat von errpt - Zusammenfassung

Im Folgenden sehen Sie ein Beispiel für Einträge, die vom Befehl **errpt** (ohne Optionen) angezeigt werden:

```
# errpt
ID          ZEITMARKE      T   C   RESSOURCENAME  BESCHREIBUNG
804E987A    1008161399      I   0   proc4          Zuordnung der CPU aufgehoben
8470267F    1008161299      T   S   proc4          Freigabe der CPU abgebrochen
1B963892    1008160299      P   H   proc4          CPU-Fehler vorhergesagt
#
```

- Wenn die Prozessorfreigabe aktiviert ist, folgt der Nachricht CPU-Fehler vorhergesagt immer eine Nachricht Zuordnung der CPU aufgehoben oder eine Nachricht Freigabe der CPU abgebrochen.
- Wenn die Prozessorfreigabe nicht aktiviert ist, wird nur die Nachricht CPU-Fehler vorhergesagt protokolliert. Wenn Sie die Prozessorfreigabe aktivieren, nachdem mindestens eine Nachricht CPU-Fehler vorhergesagt protokolliert wurde, wird der Freigabeprozess eingeleitet, und es wird ein Erfolgseintrag bzw. ein Fehlereintrag, falls für einen Prozessor ein Fehler gemeldet wird, im Fehlerprotokoll aufgezeichnet.

Langformat von errpt - Detaillierte Beschreibung

Im Folgenden sehen Sie das Format der Ausgabe des Befehls **errpt -a**:

- CPU_FAIL_PREDICTED

Fehlerbeschreibung: Vorhersehbarer Prozessorfehler

Dieser Fehler zeigt an, dass die Hardware festgestellt hat, dass ein Prozessor mit hoher Wahrscheinlichkeit in naher Zukunft ausfällt. Diese Nachricht wird immer protokolliert, egal ob die Prozessorfreigabe aktiviert ist oder nicht.

Detailldaten: *Nummer des physischen Prozessors, Position*

Beispiel für einen Fehlerprotokolleintrag - Langform

Kennsatz: **CPU_FAIL_PREDICTED**
ID: 1655419A

Datum/Zeit: Do Sep 30 13:42:11
Folgenummer: 53
Maschinen-ID: 00002F0E4C00
Knoten-ID: auntbea
Klasse: H
Typ: PEND
Ressourcenname: **proc25**
Ressourcenklasse: processor
Ressourcentyp: proc_rspc
Position: **00-25**

Beschreibung
CPU-Fehler vorhergesagt

Mögliche Ursachen
CPU-FEHLER

Fehlerursachen
CPU-FEHLER

Empfohlene Aktionen
SICHERSTELLEN, DASS CPU-GUARD-MODUS AKTIVIERT IST
SYSTEMDIAGNOSEPROGRAMME AUSFÜHREN

Detailldaten
FEHLERDATEN
0144 1000 0000 003A 8E00 9100 1842 1100 1999 0930 4019
0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 4942 4D00 5531
2E31 2D50 312D 4332 0000
0002 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

```

0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000
... ..

```

- CPU_DEALLOC_SUCCESS

Fehlerbeschreibung: Ein Prozessor wurde erfolgreich freigegeben, nachdem ein vorhersehbarer Prozessorfehler festgestellt wurde. Diese Nachricht wird protokolliert, wenn die Prozessorfreigabe aktiviert ist und die CPU erfolgreich freigegeben worden ist.

Datendaten: *Logische CPU-Nummer des freigegebenen Prozessors.*

Beispiel für einen Fehlerprotokolleintrag in Langform:

Kennsatz: **CPU_DEALLOC_SUCCESS**
 ID: 804E987A

Datum/Zeit: Do Sep 30 13:44:13
 Folgenummer: 63
 Maschinen-ID: 00002F0E4C00
 Knoten-ID: auntbea
 Klasse: 0
 Type: INFO
 Ressourcenname: **proc24**

Beschreibung
 CPU FREIGEgeben

Empfohlene Aktionen
 WARTUNG ERFORDERLICH AUFGRUND VON CPU-FEHLER

Datendaten
 FREIGEgebENE LOGISCHE CPU-NUMMER

0

In diesem Beispiel wurde **proc24** erfolgreich freigegeben und hatte beim Auftreten des Fehlers die logische CPU-Nummer **0**.

- CPU_DEALLOC_FAIL

Fehlerbeschreibung: Eine Prozessorfreigabe ist aufgrund eines vorhersehbaren Prozessorfehlers gescheitert. Diese Nachricht wird protokolliert, wenn die CPU-Freigabe aktiviert ist und die CPU nicht erfolgreich freigegeben werden konnte.

Datendaten: *Ursachencode, logische CPU-Nummer, weitere Informationen je nach Typ des Fehlers.*

Der Ursachencode ist ein numerischer Hexadezimalwert. Die möglichen Ursachencodes sind:

Eintrag	Beschreibung
2	Mindestens ein Prozess/Thread bleibt an die letzte logische CPU gebunden. In diesem Fall enthalten die Datendaten die PIDs der jeweiligen Prozesse.
3	Ein registrierter Treiber oder eine Kernelerweiterung hat bei Benachrichtigung einen Fehler zurückgegeben. In diesem Fall enthält das Feld mit den Datendaten den Namen des fehlerhaften Treibers bzw. der fehlerhaften Kernelerweiterung (ASCII-Codierung).
4	Die Freigabe eines Prozessors führt dazu, dass die Maschine weniger als zwei verfügbare CPUs hat. Dieses Betriebssystem gibt nicht mehr als $N-2$ Prozessoren auf einer N -Wege-Maschine frei, um Verwirrungen bei Anwendungen oder Kernelerweiterungen zu vermeiden, die die Gesamtanzahl der verfügbaren Prozessoren verwenden, um festzustellen, ob sie auf einem Einprozessorsystem, auf dem es sicher ist, die Verwendung von Mehrprozessorsperren zu übergehen, oder auf einem SMP-System (Symmetric Multi Processor) ausgeführt werden.
200 (0xC8)	Die Prozessorfreigabe ist inaktiviert (das ODM-Attribut cpuguard hat den Wert disable). Sie sehen diesen Fehler normalerweise nur, wenn Sie ha_star manuell starten.

Beispiele für Fehlerprotokolleinträge im Langformat

Beispiel 1:

Kennsatz: **CPU_DEALLOC_ABORTED**
ID: 8470267F
Datum/Zeit: Do Sep 30 13:41:10
Folgenummer: 50
Maschinen-ID: 00002F0E4C00
Knoten-ID: auntbea
Klasse: S
Typ: TEMP
Ressourcenname: **proc26**

Beschreibung
Freigabe der CPU abgebrochen

Mögliche Ursachen
Softwareprogramm

Fehlerursachen
Softwareprogramm

Empfohlene Aktionen
WARTUNG ERFORDERLICH AUFGRUND VON CPU-FEHLER
SIEHE BENUTZERDOKUMENTATION FÜR CPU-GUARD

Detaildaten
URSACHE FÜR FEHLGESCHLAGENE FREIGABE
0000 0003
DATEN FÜR FEHLGESCHLAGENE FREIGABE
6676 6861 6568 3200

In diesem Beispiel ist die Freigabe von **proc26** gescheitert. Der Ursachencode 3 bedeutet, dass eine Kernelerweiterung einen Fehler an die Benachrichtigungsroutine des Kernels zurückgegeben hat. Der Wert von DATEN FÜR FEHLGESCHLAGENE FREIGABE entspricht **fvhaeh2**. Dies ist der Name der Erweiterung, die für die Registrierung beim Kernel verwendet wurde.

Beispiel 2:

Kennsatz: **CPU_DEALLOC_ABORTED**
ID: 8470267F
Datum/Zeit: Do Sep 30 14:00:22
Folgenummer: 71
Maschinen-ID: 00002F0E4C00
Knoten-ID: auntbea
Klasse: S
Typ: TEMP
Ressourcenname: **proc19**

Beschreibung
Freigabe der CPU abgebrochen

Mögliche Ursachen
Softwareprogramm

Fehlerursachen
Softwareprogramm

Empfohlene Aktionen
WARTUNG ERFORDERLICH AUFGRUND VON CPU-FEHLER
SIEHE BENUTZERDOKUMENTATION FÜR CPU-GUARD

Detaildaten
URSACHE FÜR FEHLGESCHLAGENE FREIGABE
0000 0002
DATEN FÜR FEHLGESCHLAGENE FREIGABE
0000 0000 0000 **4F4A**

In diesem Beispiel ist die Freigabe von **proc19** gescheitert. Der Ursachencode 2 gibt an, dass Threads an den letzten logischen Prozessor gebunden waren und die Bindung nach dem Emp-

fang des Signals SIGCPUFAIL nicht aufgehoben wurde. Das Feld DATEN FÜR FEHLGESCHLAGENE FREIGABE zeigt, dass diese Threads zum Prozess **0x4F4A** gehören.

Mit den Optionen des Befehls **ps** (-o THREAD, -o BND) können Sie alle Threads oder Prozesse zusammen mit der Nummer der CPU, an die sie gebunden sind, auflisten.

Beispiel 3:

Kennsatz: **CPU_DEALLOC_ABORTED**
ID: 8470267F

Datum/Zeit: Do Sep 30 14:37:34
Folgenummer: 106
Maschinen-ID: 00002F0E4C00
Knoten-ID: auntbea
Klasse: S
Typ: TEMP
Ressourcenname: **proc2**

Beschreibung
Freigabe der CPU abgebrochen

Mögliche Ursachen
Softwareprogramm

Fehlerursachen
Softwareprogramm

Empfohlene Aktionen
WARTUNG ERFORDERLICH AUFGRUND VON CPU-FEHLER
SIEHE BENUTZERDOKUMENTATION FÜR CPU-GUARD

Detaildaten
URSACHE FÜR FEHLGESCHLAGENE FREIGABE
0000 0004
DATEN FÜR FEHLGESCHLAGENE FREIGABE
0000 0000 0000 0000

In diesem Beispiel ist die Freigabe von **proc2** gescheitert, weil beim Auftreten des Fehlers zwei oder weniger aktive Prozessoren verfügbar waren (Ursachencode 4).

Systemumgebungsvariablen konfigurieren

Die Systemumgebung setzt sich primär aus der Gruppe von Variablen zusammen, die bestimmte Aspekte der Prozessausführung definieren oder steuern.

Sie werden jedes Mal, wenn eine Shell gestartet wird, gesetzt bzw. zurückgesetzt. Aus der Perspektive der Systemverwaltung ist es wichtig sicherzustellen, dass für den Benutzer bei der Anmeldung die richtigen Werte konfiguriert werden. Die meisten dieser Variablen werden während der Systeminitialisierung gesetzt. Ihre Definitionen werden aus der Datei `/etc/profile` gelesen oder standardmäßig gesetzt.

Systembatterie testen:

Wenn Ihr System nicht mehr die richtige Uhrzeit anzeigt, ist die Batterie möglicherweise leer oder kontaktlos.

1. Geben Sie den folgenden Befehl **diag** ein, um den Status Ihrer Systembatterie zu prüfen:
`diag -B -c`
2. Wenn das Hauptmenü "Diagnose" erscheint, wählen Sie die Option **Fehlerbestimmung** aus. Wenn die Batterie kontaktlos oder leer ist, erscheint ein Fehlermenü mit einer Serviceanforderungsnummer (SRN, Service Request Number). Notieren Sie die SRN unter Punkt 4 im Fehlerbeschreibungsformular und melden Sie das Problem Ihrer Hardwareserviceorganisation.

Wenn Ihre Systembatterie betriebsbereit ist, wurde die Systemzeit möglicherweise falsch zurückgesetzt, weil der Befehl **date** oder **setclock** nicht ordnungsgemäß oder nicht erfolgreich ausgeführt wurde.

Zugehörige Konzepte:

„Systemuhr einstellen“

Die Systemuhr protokolliert die Zeit von Systemereignissen, ermöglicht Ihnen, Systemereignisse (z. B. die Ausführung der Hardwarediagnose um 3:00 nachts) zu planen, und teilt Ihnen mit, wann Dateien erstmalig und zuletzt gespeichert wurden.

Systemuhr einstellen:

Die Systemuhr protokolliert die Zeit von Systemereignissen, ermöglicht Ihnen, Systemereignisse (z. B. die Ausführung der Hardwarediagnose um 3:00 nachts) zu planen, und teilt Ihnen mit, wann Dateien erstmalig und zuletzt gespeichert wurden.

Verwenden Sie zum Einstellen der Systemuhr den Befehl **date**. Mit dem Befehl **setclock** können Sie Datum und Uhrzeit einstellen, indem Sie eine Verbindung zu einem Zeitserver herstellen.

Zugehörige Tasks:

„Systembatterie testen“ auf Seite 65

Wenn Ihr System nicht mehr die richtige Uhrzeit anzeigt, ist die Batterie möglicherweise leer oder kontaktlos.

Befehl date:

Mit dem Befehl **date** können Sie das Datum und die Uhrzeit anzeigen und festlegen.

Geben Sie den folgenden Befehl ein, um festzustellen, was Ihr System als aktuelles Datum und als aktuelle Uhrzeit erkennt.

```
/usr/bin/date
```

Achtung: Ändern Sie das Datum nicht, wenn auf dem System mehrere Benutzer arbeiten.

Die folgenden Formate können verwendet werden, wenn das Datum mit dem Parameter *Datum* eingestellt wird:

- *mmddHHMM[YYyy]* (Standardeinstellung)
- *mmddHHMM[yy]*

Die Variablen für den Parameter *Datum* sind wie folgt definiert:

Eintrag Beschreibung

<i>mm</i>	Gibt die Nummer des Monats an.
<i>dd</i>	Gibt die Nummer des Tags im Monat an.
<i>HH</i>	Gibt die Stunde des Tages (24-Stunden-Format) an.
<i>MM</i>	Gibt die Nummer der Minute an.
<i>YY</i>	Gibt die ersten zwei Stellen einer vierstelligen Jahresangabe an.
<i>yy</i>	Gibt die letzten beiden Zahlen des Jahres an.

Wenn Sie Rootberechtigung besitzen, können Sie mit dem Befehl **date** das aktuelle Datum und die aktuelle Zeit festlegen. Beispiel:

```
date 021714252002
```

Dieser Befehl setzt das Datum auf den 17. Februar 2002 und die Uhrzeit auf 14:25. Weitere Informationen zum Befehl **date** finden Sie in der zugehörigen Beschreibung in der Veröffentlichung *Commands Reference, Volume 2*.

Befehl setclock:

Der Befehl **setclock** zeigt Uhrzeit und Datum an bzw. legt diese Angaben fest, indem er die aktuelle Zeit von einem Zeitserver in einem Netz anfordert.

Geben Sie Folgendes ein, um Datum und Uhrzeit des Systems anzuzeigen:

```
/usr/sbin/setclock
```

Der Befehl **setclock** verwendet die erste Antwort des Zeitserver, konvertiert die dort gefundenen Angaben zur Kalenderuhr und zeigt das lokale Datum und die lokale Uhrzeit an. Wenn kein Zeitserver antwortet oder wenn das Netz nicht betriebsbereit ist, zeigt der Befehl **setclock** eine diesbezügliche Nachricht an und lässt Datum und Uhrzeit unverändert.

Anmerkung: Jeder Host, auf dem der Dämon **inetd** ausgeführt wird, kann als Zeitserver dienen.

Wenn Sie Rootberechtigung besitzen, können Sie mit dem Befehl **setclock** eine Anforderung an einen Zeitserverhost im Internet senden und das lokale Datum und die lokale Uhrzeit entsprechend setzen. Beispiel:

```
setclock Zeithost
```

Zeithost steht für den Hostnamen oder die IP-Adresse des Zeitserver.

Zugehörige Informationen:

```
setclock command
```

Unterstützung und Definition der Olson-Zeitzone:

Beginnend mit AIX 6.1 werden Zeitzonewerte unterstützt, die mit der Olson-Datenbank konsistent sind.

Die POSIX-Zeitzone-Spezifikation, die in früheren Releases von AIX unterstützt wurde, kann Änderungen der Zeitzone-Regeln, z. B. die Sommerzeit, nicht adäquat ausführen. Die Olson-Datenbank verwaltet einen gespeicherten Satz mit Zeitzone-Regeln. Wenn sich die Regeln in einer bestimmten Position ändern, kann AIX die Daten und Uhrzeiten sowohl in der Gegenwart als auch in der Vergangenheit ordnungsgemäß interpretieren.

Zeitzone-Definitionen, die mit der POSIX-Spezifikation übereinstimmen, werden von AIX weiterhin unterstützt und erkannt. AIX prüft die Umgebungsvariable **TZ**, um festzustellen, ob die Umgebungsvariable mit einem Olson-Zeitzone-Wert übereinstimmt. Wenn die Umgebungsvariable **TZ** nicht mit einem Olson-Zeitzone-Wert übereinstimmt, folgt AIX den Regeln der POSIX-Spezifikation.

Weitere Einzelheiten zur Umgebungsvariablen **TZ** finden Sie unter Environment file.

Zum Festlegen der Zeitzone mithilfe von definierten Olson-Werten verwenden Sie folgenden SMIT-Pfad: **Systemumgebungen > Datum, Uhrzeit und Zeitzone ändern/anzeigen > Zeitzone mit systemdefinierten Werten ändern.**

Konfiguration der Nachricht des Tages:

Die Nachricht des Tages (MOTD, Message of the Day) wird angezeigt, sobald sich ein Benutzer am System anmeldet.

Die Nachricht des Tages ist ein komfortables Mittel, um Informationen an alle Benutzer zu verteilen, z. B. die Versionsnummern installierter Software oder aktuelle Neuigkeiten zum System. Wenn Sie die Nachricht des Tages ändern möchten, editieren Sie in dem von Ihnen bevorzugten Editor die Datei `/etc/motd`.

AIX Runtime Expert

AIX Runtime Expert stellt einen einfachen Satz von Aktionen bereit, die Sie für eine einzelne Konsolidierung verwenden können, um die Laufzeitumgebung für eine oder mehrere Instanzen von AIX zu erfassen, anzuwenden und zu prüfen.

Es gibt Tools, die von Komponenten von AIX bereitgestellt werden, wie z. B. Reliability Availability Serviceability (RAS), Sicherheit oder Kernel, mit denen Sie die Einstellungen auf jeder Komponentenebene ändern können, um das Betriebssystem für einen bestimmten Bedarf oder eine bestimmte Anforderung zu optimieren. AIX Runtime Expert unterstützt eine systemweite Konfiguration durch die Verwendung eines erweiterbaren Frameworks für die Behandlung vieler verschiedener Konfigurationsmethoden, die derzeit in AIX vorhanden sind.

AIX Runtime Expert führt Konfigurationsbefehle für mehrere Komponenten über ein Konfigurationsprofil in einer einzelnen Aktion aus. Sie können dieses Profil verwenden, um dieselben Einstellungen auf mehrere Systeme anzuwenden. AIX Runtime Expert ist eine vereinfachte Alternative für die Verwaltung der Laufzeitkonfiguration eines oder mehrerer Systeme, steht aber der Verwendung anderer Methoden zum Ändern von Systemeinstellungen nicht im Wege.

Konzepte von AIX Runtime Expert

Sie müssen grundlegende Kenntnisse in AIX Runtime Expert besitzen, bevor Sie das Produkt verwenden können.

Die Basisfunktionen von AIX Runtime Expert unterstützen die Verwaltung und Anwendung von Konfigurationsprofilen für ein einzelnes AIX-System. Für die Unterstützung des skalierbaren Einsatzes mehrerer Systeme für ein einzelnes Profil, kann eine LDAP-basierte Profilbeschreibung von AIX-Systemen verwendet werden, wenn diese gestartet werden oder wenn sie über Verwaltungsoperationen an den AIX-Zielendpunkten dementsprechend angewiesen werden. Die Fernverwaltung für AIX Runtime Expert kann nur über die Komponente Network Install Manager (NIM) erfolgen. Unter Verwendung vorhandener NIM-Funktionen können Sie AIX Runtime Expert über Fernzugriff auf mehreren eigenständigen NIM-Clients von einer NIM-Mastermaschine aus ausführen.

Profile von AIX Runtime Expert:

Profile von AIX Runtime Expert werden verwendet, um Werte auf einem aktiven System festzulegen, Werte für ein aktives System zu extrahieren und Werte mit einem aktiven System oder mit einem anderen Profil zu vergleichen.

Ein Profil beschreibt ein oder mehrere Laufzeitkonfigurationssteuerelemente und deren Einstellungen für den Zielfunktionsbereich. Ein Profil kann einen vollständigen Satz von Steuerelementen oder eine Untergruppe von Steuerelementen und deren Werte darstellen. Konfigurationsprofile sind Standard-XML-Dateien. Mit AIX Runtime Expert können Sie Profile verwalten und auf das definierte System anwenden.

Ein Profil kann wie Musterprofile Konfigurationsparameter und Optimierungsparameter ohne Werte enthalten. Der Zweck eines Profils ohne Parameter ist der, die aktuellen Systemwerte aus dem angegebenen Profil zu extrahieren. Für Profile, die mindestens einen Parameter ohne Wert enthalten, gelten die folgenden Einschränkungen:

- Bei der Ausführung des Befehls **artexset** tritt ein Fehler auf.
- Bei der Ausführung des Befehls **artexdiff** wird eine Warnung für jeden Parameter zurückgegeben, der keinen Wert hat.

Der Wert eines Parameters in einem Profil kann Folgendes sein:

- Kein Wert
- Ein BLOB-Wert (großes Binärobjekt), d. h. base64-codierte Binärdateien wie in einer integrierten Textdatei. Der BLOB-Wert wird verwendet, um vorhandene Dateien wie `/etc/motd` oder `/etc/hosts` zu ersetzen.

- Ein Wert, der kein BLOB-Wert ist, ist ein Wert, der Systemkonfigurationsparametern zugeordnet wird, z. B. eine ganze Zahl oder eine Zeichenfolge.

Im Verzeichnis `/etc/security/artex/samples` finden Sie vorhandene Musterprofile. Die Musterprofile enthalten nur Parameternamen, die von den mit AIX Runtime Expert installierten Standardeinstellungen unterstützt werden. Die Parameter in den Musterprofilen haben keine Werte. Musterprofile sind schreibgeschützte Dateien. Verwenden Sie die Musterprofile als Schablone für die Erstellung neuer Konfigurationsprofile. Die vorhandenen Muster können nicht auf ein aktives System angewendet werden.

Im Folgenden sind einige Beispiele für Basiskonfigurationsbefehle aufgelistet, die über Konfigurationsprofile gesteuert werden können:

- Netzkonfiguration
 - no
 - mktcpip
- Kernelkonfiguration
 - ioo
 - schedo
- RAS-Konfiguration
 - alog
- Sicherheitskonfiguration
 - setsecattr

Beispiel

Das folgende Beispiel zeigt ein Konfigurationsprofil für verschiedene Kataloge und Unterkataloge mit zugeordneten Werten für verschiedene Parameter an: Sie können dieses Profil mit einem beliebigen XML-Editor bearbeiten. Sie können aber auch den Befehl `vi` verwenden und die vorhandenen Werte für die definierten Parameter ändern.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Profile origin="get" version="1.0" date="2009-04-25T15:33:37Z">
<Catalog id="vmoParam">
<Parameter name="kernel_heap_psize" value="0" applyType="nextboot" reboot="true" />
<Parameter name="maxfree" value="1088" />
</Catalog>
<Catalog id="noParam">
<SubCat id="tcp_network">
<Parameter name="tcp_recvspace" value="16384" />
<Parameter name="tcp_sendspace" value="16384" />
</SubCat>
<SubCat id="general_network">
<Parameter name="use_sndbufpool" value="1" applyType="nextboot" reboot="true" />
</SubCat>
</Catalog>
<Catalog id="lvmoParam">
<Parameter name="max_vg_pbuf_count" value="0">
<Target class="vg" instance="rootvg" />
</Parameter>
<Parameter name="pv_pbuf_count" value="512">
<Target class="vg" instance="rootvg" />
</Parameter>
</Catalog>
```

Zugehörige Tasks:

„Profile von AIX Runtime Expert ändern“ auf Seite 73

Profile von AIX Runtime Expert sind XML-Dateien, die mit einem beliebigen XML- oder Texteditor geändert werden können.

„Profile von AIX Runtime Expert erstellen“ auf Seite 72

Sie können vorhandene Muster im Verzeichnis `/etc/security/artex/samples` verwenden, um mit dem Befehl **artexget** ein neues Profil zu erstellen. Die Musterprofile sind eine Schablone für die Erstellung eines Profils, das Sie ändern und in einer Anpassungsdatei speichern können.

„Profilwerte von AIX Runtime Expert abrufen“ auf Seite 75

Verwenden Sie den Befehl **artexget**, um Informationen zu einem Profil zu suchen.

„Profile von AIX Runtime Expert anwenden“ auf Seite 76

Wenn Sie ein System mit den Konfigurations- und Optimierungsparametern aus einem Profil konfigurieren möchten, wenden Sie ein Profil mit dem Befehl **artexset** an.

Kataloge von AIX Runtime Expert:

Kataloge sind der Mechanismus, der die Konfigurationssteuerelemente definiert und spezifiziert, die von AIX Runtime Expert bearbeitet werden können.

Kataloge werden für die Steuerelemente bereitgestellt, die derzeit von AIX Runtime Expert unterstützt werden. Kataloge sind Definitionsdateien, die Konfigurationsprofilwerte Parametern zuordnen, die Befehle und Konfigurationsaktionen ausführen.

AIX Runtime Expert stellt Ihnen vorhandene schreibgeschützte Kataloge im Verzeichnis `/etc/security/artex/catalogs` bereit, die Werte enthalten, die geändert werden können. Ändern Sie diese Kataloge nicht.

Jeder Katalog enthält Parameter für jeweils eine einzige Komponente. Einige Kataloge können jedoch Parameter aus mehreren zusammengehörigen Komponenten enthalten. Die Namen der Kataloge beschreiben die Komponenten, die im Katalog enthalten sind. Das XML-Element `<description>` in jedem Katalog enthält eine Beschreibung des Katalogs.

AIX Runtime Expert und LDAP:

AIX Runtime Expert kann Profile vom LDAP-Server (Lightweight Directory Access Protocol) abrufen.

Die Profile von AIX Runtime Expert müssen als `ibm-artexProfile`-Objekte gespeichert werden und die folgenden verbindlichen Attribute haben:

- `ibm-artexProfileName`. Der Name des Profils von AIX Runtime Expert.
- `ibm-artexProfileXMLData`. Der XML-Inhalt des Profils von AIX Runtime Expert, der als `octetString` gespeichert wird.

Das Schema von AIX Runtime Expert muss auf dem LDAP-Server installiert werden, bevor Profile von AIX Runtime Expert gespeichert werden. Die Konfiguration eines LDAP-Servers für AIX Runtime Expert ist ähnlich wie die Konfiguration eines LDAP-Servers für die Benutzerauthentifizierung. Weitere Informationen zur Konfiguration von LDAP finden Sie im Abschnitt zur Konfiguration eines ITDS-Sicherheitsinformationsservers.

Die Konfiguration eines LDAP-Clients für AIX Runtime Expert ist ähnlich wie die Konfiguration eines LDAP-Clients für die Benutzerauthentifizierung. Weitere Informationen finden Sie unter LDAP-Client konfigurieren. Zum Konfigurieren eines LDAP-Clients verwenden Sie den Befehl **mksecldap -c**, um den Dämon **secldapclntd** ordnungsgemäß zu konfigurieren. AIX Runtime Expert stützt sich beim Zugriff auf den LDAP-Server auf den Dämon **secldapclntd**. Standardmäßig sucht AIX Runtime Expert unter der Kennung DN: `ou=artex,cn=AIXDATA` nach Profileinträgen. Sie können diesen definierten Namen anpassen, indem Sie den Schlüssel `artexbasedn` in der `secldapclntd`-Konfigurationsdatei `/etc/security/ldap/ldap.cfg` aktualisieren.

Profil von AIX Runtime Expert hochladen

Zum Hochladen eines Profils von AIX Runtime Expert können Sie eine LDIF-Datei (LDAP Data Interchange Format) erstellen und den Befehl **ldapadd** verwenden, oder Sie können ein LDAP-Webverwaltungstool wie Tivoli Directory Server Web Administration Tool verwenden.

Im Folgenden sehen Sie ein Beispiel für ein Profil, das in LDIF gespeichert ist:

```
dn: ou=artex,cn=AIXDATA
objectClass: organizationalUnit
objectClass: top
ou: artex

dn: ibm-artexProfileName=alogProfile.xml,ou=artex,cn=AIXDATA
objectClass: ibm-artexProfile
objectClass: top
ibm-artexProfileName: alogProfile.xml
ibm-artexProfileXMLData:< file:///etc/security/artex/samples/alogProfile.xml
```

Im Folgenden sehen Sie ein Beispiel für das Hochladen eines Profils mit dem Befehl **ldapadd** und einer LDIF-Musterdatei mit dem Namen "sample.ldif":

```
ldapadd -c -h <LDAP-Host> -D cn=admin -w <Kennwort> -f sample.ldif
```

Zugehörige Tasks:

„Profile von AIX Runtime Expert erstellen“ auf Seite 72

Sie können vorhandene Muster im Verzeichnis `/etc/security/artex/samples` verwenden, um mit dem Befehl **artexget** ein neues Profil zu erstellen. Die Musterprofile sind eine Schablone für die Erstellung eines Profils, das Sie ändern und in einer Anpassungsdatei speichern können.

Zugehörige Informationen:

IBM Security Directory Server

AIX Runtime Expert und RBAC:

RBAC (Role Based Access Control, rollenbasierte Zugriffssteuerung) kann verwendet werden, um Benutzern ohne Rootrechte die Ausführung der Befehle von AIX Runtime Expert zu ermöglichen.

Berechtigungen für AIX Runtime Expert

Bei der Installation der Dateigruppe **artex.base.rte** werden drei Systemberechtigungen erstellt, die jeweils verschiedene Stufen des Zugriffs auf die Funktionen von AIX Runtime Expert unterstützen:

- Die Berechtigung **aix.system.config.artex.read** ermöglicht die Ausführung der Befehle **artexlist** und **artexmerge**. Die Befehle **artexget** und **artexdiff** werden ebenfalls zugelassen, aber nur für den Abruf der Profilwerte. Die Werte können nicht vom System erfasst werden (d. h., der Befehl **artexget** kann nicht mit den Flags `-r`, `-n` und `-p` ausgeführt werden, und der Befehl **artexdiff** kann nur zwischen zwei Profilen ausgeführt werden).
- Die Berechtigung **aix.system.config.artex.get** lässt die Ausführung aller Operationen zu, die von der Berechtigung **artex.system.config.read** unterstützt werden, sowie die uneingeschränkte Ausführung der Befehle **artexget** und **artexdiff**.
- Die Berechtigung **aix.system.config.artex.set** lässt die Ausführung aller Operationen zu, die von der Berechtigung **artex.system.config.get** unterstützt werden, sowie die Ausführung des Befehls **artexset**.

Rollen in AIX Runtime Expert

AIX Runtime Expert erstellt keine neuen Rollen, aber die Dateigruppen **artex.base.rte** fügen der Rolle **SysConfig** die Berechtigung **aix.system.config.artex** hinzu. Alle Benutzer mit der Rolle **SysConfig** oder einer übergeordneten Rolle (wie z. B. **isso**) können die Befehle **artexlist**, **artexmerge**, **artexdiff**, **artexget** und **artexset** ausführen.

Einschränkungen

Aus Sicherheitsgründen ist die Verwendung der Umgebungsvariablen **ARTEX_CATALOG_PATH** auf den Rootbenutzer eingeschränkt. Benutzer ohne Rootrechte, denen das Recht zur Ausführung der Befehle von AIX Runtime Expert über RBAC erteilt wurde, können die Umgebungsvariable **ARTEX_CATALOG_PATH** nicht verwenden.

AIX Runtime Expert verwalten

AIX Runtime Expert verwendet einige einfache Befehle, um Profile zu erstellen, Profile zu ändern, Profile zu kombinieren und Profile anzuwenden.

AIX Runtime Expert konfigurieren:

AIX Runtime Expert verwendet die Konfigurationsdatei `/etc/security/artex/config/artex.conf`.

Ein Eintrag in der Konfigurationsdatei setzt sich aus dem Namen einer Konfigurationsdatei, gefolgt von einem oder mehreren Leerzeichen und einem Wert zusammen. Leerzeilen und Zeilen, die mit einem Nummernzeichen (#) beginnen, werden ignoriert.

Die folgenden Informationen werden unterstützt:

Tabelle 1. Konfigurationsoptionen

Optionen	Beschreibung
ARTEX_CATALOG_PATH	Eine durch Doppelpunkte getrennte Liste von Verzeichnissen, die nach Katalogdateien durchsucht werden. Diese Option wird mit der Umgebungsvariablen ARTEX_CATALOG_PATH überschrieben. Der Standardpfad ist <code>/etc/security/artex/catalogs</code> .
ARTEX_PROFILE_PATH	Eine durch Doppelpunkte getrennte Liste mit Verzeichnissen, die mit dem Befehl artexlist nach Profildateien durchsucht werden, wenn kein Verzeichnis angegeben wird. Diese Option wird mit der Umgebungsvariablen ARTEX_PROFILE_PATH überschrieben. Der Standardpfad ist <code>/etc/security/artex/samples</code> .
DEBUG_LOG_CATEGORY	Die Debugkategorie für die Protokolldatei. Diese Option kann wiederholt werden, wenn mehrere Debugkategorien ausgewählt werden sollen.
DEBUG_LOG_LEVEL	Die Debugstufe für die Protokolldatei (ein Wert zwischen 0 (keine Debug-Traces) und 3 (ausführliche Debug-Traces)).
MAX_CMDS	Die maximale Anzahl externer Befehle, die gleichzeitig ausgeführt werden. Von AIX Runtime Expert ausgeführte externe Befehle werden in die Warteschlange eingereiht, so dass nicht mehr als die mit MAX_CMDS festgelegte Anzahl externer Befehle gleichzeitig ausgeführt werden. Der Standardwert ist 10.

Profile von AIX Runtime Expert erstellen:

Sie können vorhandene Muster im Verzeichnis `/etc/security/artex/samples` verwenden, um mit dem Befehl **artexget** ein neues Profil zu erstellen. Die Musterprofile sind eine Schablone für die Erstellung eines Profils, das Sie ändern und in einer Anpassungsdatei speichern können.

Wenn Sie ein Profil mit allen Parametern erstellen möchten, die von AIX Runtime Expert unterstützt werden, führen Sie die folgenden Schritte aus:

1. Konfigurieren und optimieren Sie Ihr System so, dass es die gewünschten Einstellungen für ein neues Profil hat.
2. Wechseln Sie in das Verzeichnis mit den Mustern: `/etc/security/artex/samples`.
3. Führen Sie den folgenden Befehl aus, um ein neues Profil mit dem Namen `custom_all.xml` zu erstellen:


```
artexget -p all.xml > /Verzeichnis_für_neues_Profil/custom_all.xml
```

Anmerkung: Das Profil `custom_all.xml` kann verwendet werden, um weitere Systeme zu konfigurieren, die derzeit eine ähnliche Systemkonfiguration haben.

Zum Erstellen eines Profils für eine bestimmte Komponente, z. B. für die Netzoptionen, führen Sie die folgenden Schritte aus:

1. Konfigurieren und optimieren Sie Ihr System so, dass es die gewünschten Einstellungen für ein neues Profil hat.
2. Wechseln Sie in das Verzeichnis mit den Mustern: `/etc/security/artex/samples`.
3. Erstellen Sie ein neues Profil mit dem Namen `custom_no.xml` aus dem vorhandenen Musterprofil `noProfile.xml`, indem Sie den folgenden Befehl ausführen:

```
artexget -p noProfile.xml > /Verzeichnis_für_neues_Profil/custom_no.xml
```

Die neu erstellten Profile können angepasst werden, indem die Werte der Parameter in einem XML-Editor oder einem beliebigen Texteditor geändert oder entfernt werden.

Die angepassten Profile können auf einen LDAP-Server hochgeladen werden, so dass sie von mehreren AIX-Systemen aus verwendet werden können. Verwenden Sie zum Hochladen der Profile auf einen LDAP-Server die von LDAP bereitgestellten Tools.

Zugehörige Konzepte:

„AIX Runtime Expert und LDAP“ auf Seite 70

AIX Runtime Expert kann Profile vom LDAP-Server (Lightweight Directory Access Protocol) abrufen.

„Profile von AIX Runtime Expert“ auf Seite 68

Profile von AIX Runtime Expert werden verwendet, um Werte auf einem aktiven System festzulegen, Werte für ein aktives System zu extrahieren und Werte mit einem aktiven System oder mit einem anderen Profil zu vergleichen.

Zugehörige Tasks:

„Profilwerte von AIX Runtime Expert abrufen“ auf Seite 75

Verwenden Sie den Befehl **artexget**, um Informationen zu einem Profil zu suchen.

„Profile von AIX Runtime Expert anwenden“ auf Seite 76

Wenn Sie ein System mit den Konfigurations- und Optimierungsparametern aus einem Profil konfigurieren möchten, wenden Sie ein Profil mit dem Befehl **artexset** an.

Zugehörige Informationen:

`artexget` command

Profile von AIX Runtime Expert ändern:

Profile von AIX Runtime Expert sind XML-Dateien, die mit einem beliebigen XML- oder Texteditor geändert werden können.

Vom Benutzer mit dem Befehl **artexget** erstellt Profile können angepasst werden, indem die Werte der Parameter geändert bzw. zum Ändern oder Überwachen des Profils nicht mehr benötigte Parameter entfernt werden.

Führen Sie zum Ändern der Profile von AIX Runtime Expert die folgenden Schritte aus:

1. Führen Sie in dem Verzeichnis, in dem sich die Datei `custom_all.xml` befindet, die folgenden Befehle aus, um eine Kopie des Profils zu speichern:

```
cp custom_all.xml custom_all_backup.xml
```

2. Führen Sie in dem Verzeichnis, in dem sich die Datei `custom_all.xml` befindet, den folgenden Befehl aus, um das Profil zu bearbeiten:

```
vi custom_all.xml
```

Anmerkung: Sie können einen beliebigen XML- oder Texteditor verwenden.

3. Ändern Sie die Werte der Parameter, oder entfernen Sie Parameter, die zum Ändern oder Überwachen des Profils nicht mehr benötigt werden.
4. Führen Sie den folgenden Befehl aus, um sicherzustellen, dass die Profiländerungen ordnungsgemäß gespeichert wurden, indem Sie das Profil mit den aktuellen Systemeinstellungen vergleichen:

```
artexdiff -c -r custom_all.xml custom_all_backup.xml
```

Der Befehl **artexdiff** zeigt die Parameter an, die im Editor geändert wurden. Der <ersteWert> zeigt den Wert des Profils an, und der <zweiteWert> zeigt den aktuellen Systemwert an.

Zugehörige Konzepte:

„Profile von AIX Runtime Expert“ auf Seite 68

Profile von AIX Runtime Expert werden verwendet, um Werte auf einem aktiven System festzulegen, Werte für ein aktives System zu extrahieren und Werte mit einem aktiven System oder mit einem anderen Profil zu vergleichen.

Zugehörige Tasks:

„Profilwerte von AIX Runtime Expert abrufen“ auf Seite 75

Verwenden Sie den Befehl **artexget**, um Informationen zu einem Profil zu suchen.

„Profile von AIX Runtime Expert anwenden“ auf Seite 76

Wenn Sie ein System mit den Konfigurations- und Optimierungsparametern aus einem Profil konfigurieren möchten, wenden Sie ein Profil mit dem Befehl **artexset** an.

Zugehörige Informationen:

artexdiff command

Profile von AIX Runtime Expert kombinieren:

Ein Profil kann einen vollständigen Satz von Steuerelementen oder eine Untergruppe von Steuerelementen darstellen. Eine weitere hilfreiche Methode für die Änderung von Profilen ist die Kombination von Profilen, die eine Untergruppe von Steuerelementen darstellen, mit dem Befehl **artxmerge**.

Sie können den Befehl **artxmerge** verwenden, um mehrere Profile zu einem einzigen Profil zu kombinieren.

Führen Sie zum Kombinieren von Profilen die folgenden Schritte aus:

1. Führen Sie in dem Verzeichnis, in dem die Profile gespeichert sind, den folgenden Befehl aus:

```
artxmerge Profilname1.xml Profilname2.xml > Name_des_neuen_Profils.xml
```
2. Führen Sie den folgenden Befehl aus, um das Profil anzuzeigen und sicherzustellen, dass das Profil gültig ist:

```
artexget Name_des_neuen_Profils.xml
```

Anmerkung: Wenn die Profile, die Sie kombinieren, Parameter doppelt enthalten, schlägt die Kombination der Profile fehl. Wenn Sie jedoch das Flag **-f** verwenden, werden die Parameterwerte aus dem neuesten Profil verwendet.

Zugehörige Informationen:

artxmerge command

Profile von AIX Runtime Expert suchen:

Verwenden Sie den Befehl **artxlist**, um Profile in einem bestimmten Pfad und auf einem LDAP-Server zu suchen.

Führen Sie zum Suchen von Profilen die folgenden Schritte aus:

1. Wenn sich das Profil auf einem lokalen System befindet, führen Sie den folgenden Befehl aus:

```
artxlist
```

2. Wenn sich das Profil auf einem LDAP-Server befindet, führen Sie den folgenden Befehl aus:

```
artexlist -l
```

Standardmäßig listet der Befehl die Profile im Verzeichnis `/etc/security/artex/samples` auf. Wenn Sie den Standardpfad mit einer Umgebungsvariablen überschreiben möchten, setzen Sie `ARTEX_PROFILE_PATH` auf eine durch Semikola getrennte Liste mit Pfaden oder auf einen Pfad, der als Argument übergeben werden kann.

Zugehörige Informationen:

artexlist command

Profilwerte von AIX Runtime Expert abrufen:

Verwenden Sie den Befehl `artexget`, um Informationen zu einem Profil zu suchen.

Mit einem Profil können Sie die Werte aus dem Profil bzw. vom System in verschiedenen Formaten (XML, CSV oder Text) mit verschiedenen Filtern, z. B. Parametern, die einen Warmstart erfordern, oder Parametern, die das Stoppen und erneute Starten von Services erfordern, anzeigen.

Das Abrufen von Werten vom System ist in den folgenden Situationen hilfreich:

Momentaufnahme eines Systems erstellen

Wenn ein System ordnungsgemäß konfiguriert ist, können Sie die Konfiguration des Systems sichern, indem Sie eine Momentaufnahme erstellen. Sie können diese Momentaufnahme später verwenden, wenn Parameter geändert werden, Sie sich aber dann nicht mehr daran erinnern können, um welche Parameter es sich dabei gehandelt hat. Das Momentaufnahmenprofil kann verwendet werden, um das System auf die gewünschte Konfiguration zurückzusetzen.

Konfiguration eines Systems für die Verwendung auf anderen Systemen klonen

Nachdem Sie ein System in einer Umgebung konfiguriert und optimiert haben, können Sie die Systemeinstellungen in ein Profil von AIX Runtime Expert extrahieren und das Profil dann auf andere Systeme anwenden.

Problembhebung

Wenn auf einem Produktionssystem ein Problem auftritt, können Sie ein Profil verwenden, um dieselben Systemeinstellungen auf einem Testsystem zu konfigurieren und das Problem anschließend auf dem Testsystem zu diagnostizieren.

Führen Sie die folgenden Schritte aus, um Informationen zu einem Profil abzurufen:

1. Wechseln Sie in das Verzeichnis, in dem sich das Profil befindet, zu dem Sie Informationen abrufen möchten.
2. Führen Sie den folgenden Befehl aus, um Informationen zum Profil abzurufen:

```
artexget Name_des_Profils.xml
```

Einschränkung: Wenn auf einem System viele Benutzer definiert sind, dauert die Ausführung der Befehle `artexget`, `artexset` und `artexdiff` von AIX Runtime Expert, die auf Profile wie `chuserProfile.xml`, `coreProfile.xml` oder `all.xml` angewendet werden, länger als gewöhnlich.

Zugehörige Konzepte:

„Profile von AIX Runtime Expert“ auf Seite 68

Profile von AIX Runtime Expert werden verwendet, um Werte auf einem aktiven System festzulegen, Werte für ein aktives System zu extrahieren und Werte mit einem aktiven System oder mit einem anderen Profil zu vergleichen.

Zugehörige Tasks:

„Profile von AIX Runtime Expert erstellen“ auf Seite 72

Sie können vorhandene Muster im Verzeichnis `/etc/security/artex/samples` verwenden, um mit dem Befehl `artexget` ein neues Profil zu erstellen. Die Musterprofile sind eine Schablone für die Erstellung ei-

nes Profils, das Sie ändern und in einer Anpassungsdatei speichern können.

„Profile von AIX Runtime Expert ändern“ auf Seite 73

Profile von AIX Runtime Expert sind XML-Dateien, die mit einem beliebigen XML- oder Texteditor geändert werden können.

Zugehörige Informationen:

artexget command

Profile von AIX Runtime Expert anwenden:

Wenn Sie ein System mit den Konfigurations- und Optimierungsparametern aus einem Profil konfigurieren möchten, wenden Sie ein Profil mit dem Befehl **artexset** an.

Führen Sie die folgenden Schritte aus, um ein vom Benutzer erstelltes Profil anzuwenden:

1. Wechseln Sie in das Verzeichnis, in dem das Profil gespeichert ist, das Sie anwenden möchten.
2. Führen Sie den folgenden Befehl aus, um das Profil auf das System anzuwenden:
`artexset -c Name_des_Profils.xml`
3. Optional: Wenn Sie möchten, dass ein Profil jedes Mal angewendet wird, wenn das System erneut gestartet wird, um eine konsistente Konfiguration zu gewährleisten, führen Sie den folgenden Befehl aus:
`artexset -b Name_des_Profils.xml`

Anmerkung: Die eingeschränkten Parameter werden als schreibgeschützte Parameter unterstützt. Deshalb können die Werte dieser Parameter zwar mit dem Befehl **artexget** abgerufen, aber nicht mit dem Befehl **artexset** definiert werden.

Zugehörige Konzepte:

„Profile von AIX Runtime Expert“ auf Seite 68

Profile von AIX Runtime Expert werden verwendet, um Werte auf einem aktiven System festzulegen, Werte für ein aktives System zu extrahieren und Werte mit einem aktiven System oder mit einem anderen Profil zu vergleichen.

Zugehörige Tasks:

„Profile von AIX Runtime Expert erstellen“ auf Seite 72

Sie können vorhandene Muster im Verzeichnis `/etc/security/artex/samples` verwenden, um mit dem Befehl **artexget** ein neues Profil zu erstellen. Die Musterprofile sind eine Schablone für die Erstellung eines Profils, das Sie ändern und in einer Anpassungsdatei speichern können.

„Profile von AIX Runtime Expert ändern“ auf Seite 73

Profile von AIX Runtime Expert sind XML-Dateien, die mit einem beliebigen XML- oder Texteditor geändert werden können.

Zugehörige Informationen:

artexset command

Rollback-Operation für AIX Runtime Expert-Profile durchführen:

Verwenden Sie den Befehl **artexset -u**, um die Konfigurationseinstellungen auf die vorherigen Konfigurationseinstellungen eines Systems zurückzusetzen. Sie können die Systemeinstellungen anwenden, die vor der Anwendung des Profils verwendet wurden.

Der Befehl **rollback** kann nicht verwendet werden, wenn Sie die Systemeinstellungen während der aktuellen Sitzung nicht geändert haben.

Ein Rollback wird nicht als erneute Erstellung eines Betriebssystemimage betrachtet. Wenn Sie den Befehl **rollback** verwenden, löschen und erstellen Sie keine Ressourcen, sondern setzen die Laufzeitkonfigurationswerte auf die vorherigen Einstellungen des Systems zurück. Mit dem Befehl **rollback** ist es nicht möglich, ein Rollback der Konfigurationswerte auf die Einstellungen durchzuführen, die zu einer bestimmten

Uhrzeit oder an einem bestimmten Datum gültig waren. Sie können die Werte nur auf die Einstellungen zurücksetzen, die vom System vor der Änderung verwendet wurden.

Der Befehl **rollback** kann in den folgenden Fällen verwendet werden:

- Auf einem System vorgenommene Konfigurationsänderungen testen. Wenn die neue Konfiguration nicht ordnungsgemäß funktioniert, können Sie die Einstellungen schnell auf eine vorherige zuverlässige Konfiguration zurücksetzen.
- Debugging für ein System durchführen. Wenn ein System nicht ordnungsgemäß funktioniert, können Sie durch ein Rollback unter Umständen prüfen, ob vorgenommene Konfigurationsänderungen zu einem neu festgestellten Problem beitragen.
- Implementierung eines neuen Profils zur Bewältigung einer speziellen Ausnahmesituation. Angenommen, eine bestimmte Aktion findet nur einmal im Monat auf dem System statt, und nach der Ausführung dieser Aktion möchten Sie die vorherige Konfiguration des Systems wiederherstellen.

Führen Sie die folgenden Schritte aus, um ein Rollback des Systems auf die vorherigen Einstellungen durchzuführen:

1. Führen Sie zum Durchführen eines Profilrollbacks den folgenden Befehl aus:

```
artexset -u
```

2. Führen Sie den folgenden Befehl aus, um durch einen Vergleich mit den Systemeinstellungen sicherzustellen, dass die Rollback-Operation erfolgreich verlaufen ist:

```
artexdiff -f txt -r -Profilname.xml
```

Anmerkung: *Profilname.xml* steht für den Namen des zuletzt auf das System angewendeten Profils.

Die Unterschiede zwischen den Systemeinstellungen und den Profileinstellungen werden angezeigt.

Zugehörige Informationen:

artexget command

artexlist command

Profile von AIX Runtime Expert vergleichen:

Verwenden Sie den Befehl **artexdiff**, um zwei Profile oder Profilwerte mit Systemwerten zu vergleichen.

Führen Sie die folgenden Schritte aus, um die Profile für zwei verschiedene Systeme zu vergleichen:

1. Führen Sie auf System 1 den folgenden Befehl aus:

```
artexget -p all.xml > all_system1.xml
```

2. Führen Sie auf System 2 den folgenden Befehl aus:

```
artexget -p all.xml > all_system2.xml
```

Wenn Sie prüfen möchten, ob sich Konfigurationsparameter auf einem System geändert haben, z. B., wenn Sie nach Ihrem Urlaub zurückkehren, führen Sie die folgenden Befehle aus.

- Führen Sie nach Ihrer Rückkehr aus dem Urlaub den folgenden Befehl aus:
- ```
$ artexget -p all.xml > all_before_vacation.xml
```
- Zum Anzeigen der Konfigurationsänderungen, die während Ihres Urlaubs vorgenommen wurden, führen Sie den folgenden Befehl aus:

```
$ artexdiff -c -p all_before_vacation.xml
```

#### **Zugehörige Informationen:**

artexget command

artexlist command

## Profile für AIX Runtime Expert schreiben

Sie können den Geltungsbereich von AIX Runtime Expert erweitern, indem Sie Kataloge und Profile hinzufügen, die das Programm verwenden kann. Sie müssen mit den Konzepten von AIX Runtime Expert vertraut sein, bevor Sie versuchen, neue Kataloge zu schreiben.

Die kleinste Einzelinformation, die von AIX Runtime Expert bearbeitet wird, ist ein Parameter. Parameter können optimierbare Parameter, Konfigurationsdateien, Umgebungsvariablen, Eigenschaften von Objekten wie Benutzer, Einheiten und Subsysteme sein (solche Objekte sind aufgerufene Ziele im Kontext von AIX Runtime Expert).

Parameter werden entsprechend der Aktivitätsdomäne (z. B. `user`, `tcpi`) in Profilen zusammengefasst. Profile sind als Mittel der Interaktion zwischen den Benutzer und dem Framework AIX Runtime Expert bestimmt. Profile sind die Eingabe für den Befehl **artexget**, der den Parameterwert auf dem System abrufen und ein Profil zurückgibt. Profile (einschließlich Werten) sind die Eingabe für den Befehl **artexset**, der die Parameter auf die in das Profil eingelesenen Werte setzt.

### Konzepte beim Schreiben eines Profils für AIX Runtime Expert:

Profile von AIX Runtime Expert sind XML-Dateien, die eine Liste mit Konfigurationsparametern und optional die Parameterwerte und Nutzungsflags enthalten.

Profile können sich auf dem zu optimierenden System befinden, wenn Sie den Befehl AIX Runtime Expert direkt in der Befehlszeile ausführen.

#### *Profilpositionen:*

Beispielprofile für AIX Runtime Expert sind im Verzeichnis `/etc/security/artex/samples` enthalten.

Wenn Sie einen neuen Katalog schreiben, den AIX Runtime Expert unterstützen soll, wird empfohlen, ein Beispielprofil zu schreiben, das als Eingabe für den Befehl **artexget** verwendet werden kann. Ein Beispielprofil ist ein schreibgeschütztes Profil, in dem den Parametern keine Werte zugeordnet sind. Vorhandene Beispielprofile sind im Verzeichnis `/etc/security/artex/samples` enthalten. Standardmäßig listet der Befehl **artexlist** nur die im Standardverzeichnis enthaltenen Profile auf, aber das Standardverzeichnis kann mit der Umgebungsvariablen **ARTEX\_PROFILE\_PATH** geändert werden. Mit dieser Umgebungsvariablen können mehrere Verzeichnisse angegeben werden, die dann mit dem Trennzeichen `:` voneinander getrennt werden müssen.

Alle Profile aus dem Verzeichnis "samples" werden während der Installation der Dateigruppe **artex.base.samples** zum Profil **default.xml** zusammengeführt, das vom Befehl **snap** verwendet wird. Ein Profil, das nicht in das Profil **default.xml** aufgenommen werden soll, darf nicht im Verzeichnis "samples" abgelegt werden. Beispiele für Profile, die nicht in das Profil **default.xml** eingeschlossen werden sollen, sind die Profile, die potenziell Tausende von Parametern enthalten können (beispielsweise, wenn das Profil Benutzer als Zielklasse verwendet), und Profile, die nur auf bestimmten Systemen ausgeführt werden sollen (z. B. das Attributprofil **vios**).

#### *Profilbenennung:*

Profile von AIX Runtime Expert werden basierend auf den Befehlen benannt.

Profile werden gewöhnlich für einen einzigen Befehl oder für eine Gruppe von Befehlen erstellt. Profile können mehrere Kataloge enthalten, wenn die Kataloge eng miteinander verwandt sind. Gemäß Konvention werden die Dateien nach dem jeweiligen Befehl benannt, z. B. **commandProfile.xml** für das Beispielprofil oder **commandParam.xml** für den Katalog, aber dies ist nicht verbindlich. Nur die Dateierweiterung **.xml** ist erforderlich.

## Profilprozess:

Beschreibt den Prozess für das Schreiben eines neuen Profils für AIX Runtime Expert.

Sie müssen die folgenden Schritte ausführen, um ein neues Profil für AIX Runtime Expert zu schreiben:

1. Erstellen Sie eine Liste der Parameter, die Sie in das Profil aufnehmen möchten.
2. Erstellen Sie ein Element `<Parameter name="...">` für jeden der Parameter, und setzen Sie das Attribut `name` auf den Namen, der im Element `<ParameterDef>` in der Katalogdatei verwendet wird.
3. Gruppieren Sie alle Parameter, die in derselben Katalogdatei definiert sind, in demselben Element `<Catalog id="...">`, und setzen Sie das Attribut `id` auf die ID, die im Element `<Catalog>` in der Katalogdatei verwendet wird.
4. Führen Sie für jedes Element `<Parameter>` die folgenden Aktionen aus:
  - a. Wenn der Parameter mit `reboot=true` in der Katalogdatei definiert ist, fügen Sie die Attribute `reboot=true` und `applyType=nextboot` hinzu.
  - b. Wenn der Parameter nur erfasst und nicht gesetzt werden muss, fügen Sie das Attribut `readOnly=true` hinzu.
  - c. Wenn der Parameter mit einem nicht leeren Attribut `targetClass` in der Katalogdatei definiert ist, gehen Sie wie folgt vor:
    - 1) Wenn eine Zielerkennung für diesen Parameter gewünscht ist, definieren Sie ein einziges Element `<Parameter>` für diesen Parameter, und verwenden Sie das Sonderziel `<Target class="" instance="">` für dieses Element.
    - 2) Wenn bestimmte Ziele für diesen Parameter definiert werden müssen, definieren Sie für jedes Ziel ein Element `<Parameter>`. Definieren Sie unter jedem Element `<Parameter>` die entsprechenden Elemente `<Target class="..." instance="..." />`, um das Ziel vollständig anzugeben.
5. Testen Sie das Profil mit dem Befehl `artexget -r`.

## Profilelemente in AIX Runtime Expert:

### Element `<Profile>`:

Das Element `<Profile>` ist das Stammelement für alle Profildateien.

## Syntax

Die folgenden Attribute werden unterstützt:

Tabelle 2. Attribute

| Attribut        | Erforderlich | Typ      | Beschreibung                                                                                              |
|-----------------|--------------|----------|-----------------------------------------------------------------------------------------------------------|
| <i>origin</i>   | Nein         | string   | Der Ursprung des Profils.                                                                                 |
| <i>date</i>     | Nein         | dateTime | Das Erstellungsdatum bzw. das Datum der letzten Änderung des Profils. Das Format ist YYYY-MM-TTThh:mm:ss. |
| <i>readOnly</i> | Nein         | boolean  | Gibt an, ob dieses Profil in einer set-Operation verwendet werden kann. Der Standardwert ist false.       |
| <i>version</i>  | Nein         | string   | Die Versionsnummer des Profils.                                                                           |

Die folgenden untergeordneten Elemente werden unterstützt:

Table 3. Untergeordnete Elemente

| Untergeordnetes Element | Erforderlich | Nummer | Beschreibung                                                                          |
|-------------------------|--------------|--------|---------------------------------------------------------------------------------------|
| <ShortDescription>      | Nein         | 0-1    | Eine Kurzbeschreibung des Katalogs.                                                   |
| <Description>           | Nein         | 0-1    | Eine lange Beschreibung des Katalogs.                                                 |
| <Comments>              | Nein         | 0-1    | Vom Benutzer bereitgestellte Kommentare.                                              |
| <Catalog>               | Nein         | 0-n    | Der Katalog, der für die Bearbeitung von Operationen für ein Profil erforderlich ist. |

## Attribute

### *origin*

Das Attribut *origin* ist ein Informationsattribut, dem die folgenden Werte zugeordnet werden können:

- Beim Erstellen eines Profils muss das Attribut *origin* auf *reference* gesetzt werden.
- Wenn ein Profil mit dem Befehl **artexget** erstellt wird, wird das Attribut *origin* automatisch auf *get* gesetzt.

## Untergeordnete Elemente

Das Element <Comments> ist eine optionale Textzeichenfolge, die für andere Zwecke reserviert ist. Dieses Element darf nicht verwendet werden, wenn ein Profil manuell erstellt wird, und es wird nicht von den Basisbefehlen von AIX Runtime Expert verwendet.

## Beispiele

1. Im Folgenden sehen Sie ein leeres Beispielprofil:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Profile origin="reference" version="2.0.0" readOnly="true">
</Profile>
```

2. Der Befehl **artexget -r /etc/security/artex/samples/smtctmProfile.xml** gibt ein Profil aus, das dem folgenden Beispiel gleicht:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Profile origin="get" version="2.0.1" date="2010-09-29T07:50:56Z">
 <Catalog id="smtctlParam" version="2.0">
 <Parameter name="enableSMT" value="1"/>
 </Catalog>
</Profile>
```

## Zugehörige Informationen

Element <Catalog>

Elemente <Description> und <ShortDescription>

Elemente <Description> und <ShortDescription>:

Die Elemente <Description> und <ShortDescription> können verwendet werden, um Beschreibungen für Profile und Parameter bereitzustellen.



## Syntax

Übergeordnetes Element des Elements **<ShortDescription>**:

- Element **<Profile>**

Mögliche übergeordnete Elemente des Elements **<Description>**:

- Element **<Profile>**
- Element **<Parameter>**

Die Elemente **<Description>** und **<ShortDescription>** haben dasselbe Format. Der im Element **<Description>** enthaltene Text ist der Zeichenfolgeinhalt des XML-Tags.

## Verwendung

Beschreibungen in Profildateien werden momentan vom Framework AIX Runtime Expert nicht verwendet. Die Befehle von AIX Runtime Expert ignorieren alle Kommentare, die im Eingabeprofil enthalten sind.

## Beispiele

Im Folgenden sehen Sie ein Beispiel für die Elemente **<Description>** und **<ShortDescription>**:

```
<ShortDescription>
 Kurze Zusammenfassung des Feldinhalts.
</ShortDescription>
<Description>
Dieses Textfeld kann verwendet werden, um die Verwendung des übergeordneten Elements ausführlich anzuzeigen.
</Description>
```

## Zugehörige Informationen

Element **<Profile>**

Element **<Parameter>**

Element **<Catalog>**:

Das Element **<Catalog>** gibt den Namen der Katalogdatei an, die die Definitionen für die untergeordneten Elemente **<Parameter>** enthält.

## Syntax

Übergeordnetes Element: **<Profile>**

Die folgenden Attribute werden unterstützt:

*Tabelle 4. Attribute*

Attribut	Erforderlich	Typ	Beschreibung
<i>id</i>	Ja	string	Gibt die Katalog-ID an. Dieser Name muss auf dem System eindeutig sein.
<i>version</i>	Nein	string	Gibt die Version des Katalogs an, der für die Erstellung des Profils verwendet wird.

Die folgenden untergeordneten Elemente werden unterstützt:

Table 5. Untergeordnete Elemente

Untergeordnetes Element	Erforderlich	Nummer	Beschreibung
<Parameter>	Nein	0-n	Der im Katalog enthaltene Parameter.
<SubCat>	Nein	0-n	Die im Katalog enthaltene Unterkategorie.
<Seed>	Nein	0-n	Der im Katalog enthaltene Seed.

## Attribute

### *id*

Das Attribut *id* muss auf den Namen des Katalogs gesetzt werden, der die im Element <Catalog> aufgelisteten Parameter definiert. Das Attribut *id* ist der Basisname der Katalogdatei auf der Platte ohne die Erweiterung *.xml*. Ein Profil verwendet beispielsweise das Element <Catalog *id="commandParam">*, um auf die Katalogdatei *commandParam.xml* zu verweisen.

Standardmäßig werden die Katalogdateien im Verzeichnis */etc/security/artex/catalogs* gesucht. Der Rootbenutzer hat jedoch als einziger die Möglichkeit, weitere Verzeichnisse zu durchsuchen, indem er die Umgebungsvariable *ARTEX\_CATALOG\_PATH* setzt. Mit dieser Umgebungsvariablen können mehrere Verzeichnisse angegeben werden, die dann mit dem Trennzeichen *:* voneinander getrennt werden müssen.

### *version*

Das Attribut *version* wird als *MM.mm* geschrieben, wobei *MM* für die Hauptversionsnummer und *mm* für die Nebenversionsnummer steht.

Das Attribut *version* muss mit der Version der referenzierten Katalogdatei übereinstimmen (siehe die Beschreibung des Elements <Catalog> im Abschnitt Kataloge von AIX Runtime Expert schreiben). Wenn ein Befehl von AIX Runtime Expert für ein Profil ausgeführt wird, das auf einen Katalog mit einer ungültigen Version verweist, wird die folgende Warnung angezeigt:

```
0590-218 Die Katalogversion weicht von der im Profil referenzierten Version ab.
Die Version des lokalen Katalogs ist '2.1'. Die zum Erstellen des Profils verwendete Version
ist '2.0'
```

## Verwendung

Das Element <Catalog> identifiziert die Katalogdatei, die die Definition der aufgelisteten Seeds und Parameter enthält. Alle Seeds und Parameterelemente in einem Profil müssen im entsprechenden Element <Catalog> enthalten sein.

Ein Profil kann auf mehrere Kataloge verweisen. Das Profil *default.xml* wird beispielsweise während der Installation der Dateigruppe *artex.base.sample* erstellt, indem eine ausgewählte Gruppe anderer Beispiele zusammengeführt wird.

## Beispiele

Das Sicherheitsattributprofil *secattrProfile.xml* verwendet drei Kataloge, die jeweils eine der Sicherheitstabellen bearbeiten:

```
<Profile origin="reference" readOnly="true" version="2.0.0">
 <Catalog id="privcmdParam" version="2.0">
 <Parameter name="privatecommands" />
 </Catalog>
 <Catalog id="privdevParam" version="2.0">
 <Parameter name="privatedevices"/>
 </Catalog>
```

```

<Catalog id="privfileParam" version="2.0">
 <Parameter name="privatefiles" />
</Catalog>
</Profile>

```

## Zugehörige Informationen

Element **<Catalog>** (in Katalogdateien)

Element **<SubCat>**:

Das Element **<SubCat>** ist ein Mittel für die Erstellung logischer Unterkategorien im Element **<Catalog>**.

## Syntax

Übergeordnetes Element: **<Catalog>**, **<SubCat>**

Die folgenden Attribute werden unterstützt:

*Tabelle 6. Attribute*

Attribut	Erforderlich	Typ	Beschreibung
<i>id</i>	Ja	string	Gibt den Unterkategorienamen an. Dieser Name muss in einem Element <b>&lt;Catalog&gt;</b> eindeutig sein.

Die folgenden untergeordneten Elemente werden unterstützt:

*Tabelle 7. Untergeordnete Elemente*

Untergeordnetes Element	Erforderlich	Nummer	Beschreibung
<b>&lt;Parameter&gt;</b>	Nein	0-n	Enthält einen Parameternamen.
<b>&lt;SubCat&gt;</b>	Nein	0-n	Verschachtelte Unterkategorie

## Attribute

Das Attribut *id* identifiziert eine Unterkategorie in einem Katalog eindeutig. Ein Profil kann mehrere Unterkategorien mit derselben ID enthalten, sofern diese nicht in demselben Element **<Catalog>** verwendet werden.

## Untergeordnete Elemente

Ein Element **<SubCat>** kann ein anderes Element **<SubCat>** als untergeordnetes Element enthalten. Die Anzahl verschachtelter definierbarer Unterkategorien ist nicht begrenzt.

## Verwendung

Unterkategorien werden nur zur besseren Lesbarkeit eingefügt. Sie haben keine Auswirkung auf die Verarbeitung der Parameter.

## Beispiele

Das Profil `noProfile.xml` enthält mehrere Unterkategorien. Im Folgenden sehen Sie ein Beispiel:

```

<Profile origin="reference" readOnly="true" version="2.0.0">
 <Catalog id="noParam" version="2.0">
 <SubCat id="general_network"
 <Parameter name="fasttimo"/>

```

```

 <Parameter name="nbc_limit"/>
 </SubCat>
 <SubCat id="tcp_network">
 <Parameter name="clean_partial_conns"/>
 <Parameter name="delayack"/>
 </SubCat>
 <SubCat id="restricted">
 <Parameter name="extendednetstats" readOnly="true"/>
 <Parameter name="inet_stack_size" readOnly="true"/>
 </SubCat>
</Catalog>
</Profile>

```

## Zugehörige Informationen

Element `<Parameter>`

*Element <Parameter>:*

Das Element `<Parameter>` definiert einen Konfigurationsparameter.

## Syntax

Die folgenden Attribute werden unterstützt:

*Tabelle 8. Attribute*

Attribut	Erforderlich	Typ	Beschreibung
<i>name</i>	Ja	string	Gibt den Namen des Parameters an. Dieser Name muss in einem Katalog eindeutig sein.
<i>value</i>	Nein	string	Der Wert des Parameters, sofern definiert.
<i>applyType</i>	Nein	string	Gibt an, ob der runtime- oder nextboot-Wert des Parameters abgerufen oder gesetzt werden muss, wenn kein Flag angegeben wird. Der Standardwert ist runtime.
<i>reboot</i>	Nein	boolean	Wenn dieses Attribut auf true gesetzt wird, ist ein Warmstart erforderlich, damit die Änderung des Werts wirksam wird. Der Standardwert ist false.
<i>readOnly</i>	Nein	boolean	Gibt an, ob der Wert des Parameters nicht gesetzt werden darf. Der Standardwert ist false.
<i>disruptive</i>	Nein	boolean	Gibt an, ob die zum Setzen des Parameters verwendete Methode Unterbrechungsbedingungen impliziert. Der Standardwert ist false.
<i>setDiscover</i>	Nein	boolean	Gibt an, ob die set-Methode auf den Wert für alle erkannten Instanzen einer Zielklasse gesetzt werden muss. Der Standardwert ist false.

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 9. Untergeordnete Elemente

Untergeordnetes Element	Erforderlich	Nummer	Beschreibung
<Value>	Nein	0-1	Der Wert des Parameters.
<Target>	Nein	0-n	Ziel, für den der Parameter gilt.
<Description>	Nein	0-1	Beschreibung des Parameters
<Property>	Nein	0-n	Eigenschaft des Parameters

## Attribute

Tabelle 10. Attribute

Attribut	Beschreibung
<i>name</i>	Der Name des Parameters ist das einzige erforderliche Attribut des Elements <Parameter>. Zusammen mit dem im übergeordneten Element <Catalog> angegebenen Katalognamen identifiziert der Parametername eine Parameterdefinition in der Katalogdatei eindeutig.
<i>value</i>	Der Wert des Parameters kann als Attribut oder als untergeordnetes Element angegeben werden.
<i>applyType</i>	<p>Das Attribut <i>applyType</i> akzeptiert die Werte <i>runtime</i> (Standardwert) und <i>nextboot</i>. Dieses Attribut bestimmt den Befehl, der zur Bearbeitung des Parameters verwendet wird.</p> <p>Für set-Operationen gibt <i>applyType=runtime</i> an, dass der Befehl &lt;Set type="permanent"&gt; aus der Katalogdatei zum Setzen des Parameters verwendet werden soll. <i>applyType=nextboot</i> gibt an, dass der Befehl &lt;Set type="nextboot"&gt; verwendet werden soll.</p> <p>Wenn für get-Operationen das Flag <i>-p</i> verwendet wird, gibt <i>applyType=runtime</i> an, dass der Befehl &lt;Get type="current"&gt; aus der Katalogdatei zum Abrufen des Parameters verwendet werden soll. &lt;applyType&gt;=nextboot gibt an, dass der Befehl &lt;Get type="nextboot"&gt; verwendet werden muss.</p> <p>Das Attribut <i>applyType</i> muss auf <i>nextboot</i> gesetzt werden, wenn das Attribut <i>reboot</i> auf "true" gesetzt ist.</p>
<i>reboot</i>	<p>Dieses Attribut hat den Standardwert "false". Wenn Sie dieses Attribut auf "true" setzen, muss das System erneut gestartet werden, damit Parameteränderungen wirksam werden. Dieses Attribut muss dem Attribut <i>reboot</i> entsprechen, das im entsprechenden Element &lt;ParameterDef&gt; der Katalogdatei definiert ist.</p> <p>Wenn Sie dieses Attribut auf "true" setzen, muss das Attribut <i>applyType</i> auf <i>nextboot</i> gesetzt werden.</p> <p>Wenn Sie einen Parameter setzen, in dem das Attribut <i>reboot</i> gesetzt ist, wird eine Warnung angezeigt, in der der Benutzer darauf hingewiesen wird, dass eine Warmstartoperation erforderlich ist:</p> <p>0590-206 Es ist eine manuelle nachbereitende Operation erforderlich, damit die Änderungen wirksam werden. Führen Sie einen Warmstart des Systems durch.</p> <p>Setzen Sie das Attribut <i>reboot</i> nur dann auf "true", wenn eine Änderung des Parameters erst nach dem nächsten Warmstart wirksam wird.</p>

Tabelle 10. Attribute (Forts.)

Attribut	Beschreibung
readOnly	<p>Dieses Attribut gibt an, dass der Wert des Parameters vom Befehl <b>artexget</b> gelesen werden kann, aber weder mit dem Befehl <b>artexset</b> gesetzt noch bei einer Vergleichsoperation mit Livewerten in Befehlen <b>artexdiff</b> berücksichtigt wird. Der Standardwert ist "false".</p> <p>Im Folgenden sind einige Situationen beschrieben, die die Definition des Attributs <i>readOnly</i> mit dem Wert "true" erfordern:</p> <ul style="list-style-type: none"> <li>• Der Parameter ist statisch, und der Parameterwert kann nicht geändert werden (z. B. der Parameter <i>memory_frames</i> im Befehl <b>vmo</b>).</li> <li>• Der Zugriff auf den Parameter ist eingeschränkt, und es davon abgeraten, dass Benutzer ihn in automatischen Prozeduren ändern. In diesem Fall müssen die set-Konfigurationsmethoden für diesen Parameter in der Katalogdatei definiert werden, aber ein Systemadministrator muss das Attribut <i>readOnly</i> manuell aus dem Profil entfernen, um den Parameter setzen zu können.</li> </ul>
setDiscover	<p>Wenn das Attribut <i>setDiscover</i> auf "true" gesetzt ist und der Befehl <b>artexset</b> mit dem Flag <b>-d</b> aufgerufen wird, muss der Befehl <b>discover</b> aufgerufen werden, um alle Instanzen des Ziels zu erkennen und alle Instanzen auf den im Profil gespeicherten Wert zu setzen.</p> <p>Der Standardwert von <i>setDiscover</i> ist "false". Der Wert "true" macht nur Sinn, wenn für den Parameter Zielklassen in der Katalogdatei definiert sind.</p> <p>Geben Sie dieses Attribut nicht an, wenn Sie ein Beispielprofil erstellen. Fortgeschrittene Benutzer fügen dieses Attribut manuell hinzu, falls es erforderlich ist.</p>

## Weitere Attribute

Die Attribute *type* und *disruptive* sind Informationsattribute, die automatisch vom Befehl **artextget** gesetzt werden, wenn dieser mit dem Flag **-i** aufgerufen wird. Schließen Sie diese Attribute nicht ein, wenn Sie ein Beispielprofil erstellen.

## Beispiele

1. Im Folgenden sehen Sie einen Beispielauszug aus dem Beispielkatalog `vmoProfile.xml`, der die Verwendung verschiedener optionaler Attribute veranschaulicht:

```
<Profile origin="reference" readOnly="true" version="2.0.0">
 <Catalog id="vmoParam" version="2.1">
 <Parameter name="nokilluid"/>
 <Parameter name="memory_frames" readOnly="true"/>
 <Parameter name="kernel_heap_psize" reboot="true" applyType="nextboot"/>
 </Catalog>
</Profile>
```

2. Wenn Sie den Befehl **artexget -r** für das Profil aus Beispiel 1 ausführen, wird das folgende Profil angezeigt:

```
<Profile origin="get" version="2.0.1" date="2011-03-24T13:41:01Z">
 <Catalog id="vmoParam" version="2.1">
 <Parameter name="nokilluid" value="0"/>
 <Parameter name="memory_frames" value="393216" readOnly="true"/>
 <Parameter name="kernel_heap_psize" value="4096" applyType="nextboot" reboot="true"/>
 </Catalog>
</Profile>
```

## Zugehörige Informationen

Parameterwerte

Element <ParameterDef>

*Parameterwerte:*

Der Wert eines Parameters kann in einem Profil als Attribut definiert werden, wenn er kurz genug ist, bzw. als untergeordnetes Element des Elements <Parameter>.

## Verwendung

Wenn Sie ein Beispielprofil schreiben, muss den Parametern kein Wert zugeordnet werden. Der Wert eines Parameters, sofern vorhanden, wird automatisch in das Profil eingeschlossen, indem ein Befehl **artexget** ausgeführt wird.

## Laufzeit- und nextboot-Werte

Das Konzept von runtime- und nextboot-Werten ist ein wichtiger Teil des Frameworks AIX Runtime Expert.

Der runtime-Wert des Parameters ist der aktuelle Wert, der vom System bei der Ausführung des Befehls **artexget** vom System abgerufen wird. Der nextboot-Wert ist der Wert, den der Parameter nach dem nächsten Warmstart des Systems annimmt.

Beispiel mit dem Parameter *type\_of\_dump* im Profil *sysdumpdevProfile.xml*: Der aktuelle (runtime-)Wert dieses Parameters kann traditionell oder von der Firmware unterstützt sein. Wenn dieser Wert geändert wird (mit dem Befehl **artexset** oder direkt mit dem Befehl **sysdumpdev**), wird er erst nach dem nächsten Warmstart wirksam. Der nextboot-Wert dieses Parameters ist dann der geänderte Wert.

```
<Parameter name="type_of_dump" applyType="nextboot" reboot="true" />
```

## Beispiel

Das folgende Beispiel zeigt einen Parameter mit einem Wert, der als Attribut angegeben wird, und einen weiteren Parameter mit einem Wert, der als untergeordnetes Element angegeben wird:

```
<Profile origin="get" version="2.0.1" date="2010-09-28T12:30:03Z">
<Catalog id="login.cfgParam" version="2.0">
<Parameter name="shells">
<Value>
/bin/sh,/bin/bsh,/bin/csh,/bin/ksh,/bin/tsh,
/bin/ksh93,/usr/bin/sh,/usr/bin/bsh,/usr/bin/csh,
/usr/bin/ksh,/usr/bin/tsh,/usr/bin/ksh93,
/usr/bin/rksh,/usr/bin/rksh93,
/usr/sbin/uucp/uucico,/usr/sbin/sliplogin,
/usr/sbin/snappd
</Value>
</Parameter>
<Parameter name="maxlogins" value="32767"/>
</Catalog>
</Profile>
```

*Element <Property>:*

Das Element <Property> weist einer Parametereigenschaft einen Wert zu.

## Syntax

Übergeordnetes Element: **<Parameter>**

Die folgenden Attribute werden unterstützt:

*Tabelle 11. Attribute*

Attribut	Erforderlich	Typ	Beschreibung
<i>name</i>	Ja	string	Gibt den Namen der Eigenschaft an.
<i>value</i>	Nein	string	Gibt den Wert der Eigenschaft an.

## Verwendung

Das Element **<Property>** weist dem Eigenschaftsnamen des übergeordneten Elements einen Wert zu. Dieser Wert wird verwendet, wenn die Sequenz **%p[Name]** während der Befehlszeilengenerierung erweitert wird.

Das Element **<Property>** wird Profilen im Allgemeinen nicht manuell hinzugefügt. Das Element wird basierend auf dem im entsprechenden Element **<Property>** der Katalogdatei definierten Befehl automatisch in das Ausgabeprofil eingefügt, wenn die Befehle **artexget -r** und **artexget -n** ausgeführt werden.

## Beispiel

Im folgenden Beispiel wird die Eigenschaft **nodeId** für den Parameter **netaddr** gesetzt. Der Eigenschaftswert wird vom Befehl **artexget -r** erfasst und ist die Ausgabe des Befehls **uname -f**:

```
<Parameter name="netaddr" value="172.16.128.13">
 <Target class="device" instance="en0"/>
 <Property name="nodeId" value="8000108390E00009"/>
</Parameter>
```

## Zugehörige Informationen

„Element **<PropertyDef>**“ auf Seite 124 (in den Katalogdateien)

*Element <Seed>*:

Das Element **<Seed>** definiert einen Seed, der während der Ausführung der **<Get>**-Operation zu mindestens einem Element **<ParameterDef>** erweitert wird.

## Syntax

Übergeordnetes Element: **<Catalog>**

Das folgende Attribut wird unterstützt:



Tabelle 12. Attribut

Attribut	Erforderlich	Typ	Beschreibung
<i>name</i>	Ja	string	Gibt den Namen des Seed-Elements an, das einem Element <b>SeedDef</b> in der Katalogdatei entspricht.

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 13. Untergeordnete Elemente

Untergeordnetes Element	Erforderlich	Nummer	Beschreibung
<Parameter>	Nein	0-n	Filtert erkannte Parameter basierend auf den Namen der Parameter.
<Target>	Nein	0-n	Filtert erkannte Parameter basierend auf ihren Zielen.

## Verwendung

Das Element <Seed> erkennt Parameter dynamisch während einer <Get>-Operation.

Wenn der Befehl **artexget** ausgegeben wird, wird jedes Element <Seed> im Eingabeprofil zu einem oder mehreren Elementen <Parameter> erweitert. Die Profile werden basierend auf den Regeln erweitert, die im entsprechenden Element <SeedDef> der Katalogdatei definiert sind. Dieser Prozess wird als Parametererkennung bezeichnet. Nach Abschluss des Erkennungsprozesses führt der Befehl **artexget** wie gewöhnlich mit der Verarbeitung des erweiterten Profils fort.

Die optionalen Elemente <Parameter> und <Target> werden verwendet, um die erkannten Parameter zu filtern. Erkannte Parameter, die den im Unterelement <Parameter> definierten Kriterien nicht entsprechen, werden verworfen. Parameter, die für Ziele gelten, die den im Unterelement <Target> definierten Kriterien nicht entsprechen, werden ebenfalls verworfen.

## Beispiele

In diesem Beispiel wird der Katalog **devSeed** verwendet, um einen Seed zu definieren, der für die Erkennung aller Attribute aller Einheiten verwendet werden kann:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Catalog id="devSeed" version="3.0">
 <SeedDef name="devAttr">
 <Discover>
 <Command>
 /usr/sbin/lisdev -F 'name class subclass type' |
 while read DEV CLASS SUBCLASS TYPE
 do
 CAT=devParam.$CLASS.$SUBCLASS.$TYPE
 /usr/sbin/lisattr -F attribute -l $DEV |
 while read PAR
 do
 echo "device=$DEV $CAT $PAR"
 done
 done
 </Command>
 Mask target="1" catalog="2" name="3">(.*) (.*) (.*)</Mask>
 </Discover>
 </SeedDef>
</Catalog>
```

Das folgende Profil kann verwendet werden, um alle unterstützten Attribute aller unterstützten Einheiten zu erkennen:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Profile>
 <Catalog id="devSeed" version="3.0">
 <Seed name="devAttr"/>
 </Catalog>
</Profile>
```

Unter Verwendung desselben Attributs kann ein **<Target>**-Filter verwendet werden, um alle unterstützten Attribute aller Ethernet-Adapter zu erkennen:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Profile>
 <Catalog id="devSeed" version="3.0">
 <Seed name="devAttr">
 <Target class="device" match="^en[0-9]+$"/>
 </Seed>
 </Catalog>
</Profile>
```

Es kann ein **<Parameter>**-Filter hinzugefügt werden, um nur die Attribute **netaddr**, **netaddr6**, **alias** und **alias6** aller Ethernet-Adapter zu erfassen:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Profile>
 <Catalog id="devSeed" version="3.0">
 <Seed name="devAttr">
 <Parameter match="^(netaddr|alias)6?$"/>
 <Target class="device" match="^en[0-9]+$"/>
 </Seed>
 </Catalog>
</Profile>
```

## Zugehörige Informationen

„Element **<SeedDef>**“ auf Seite 117 (in Katalogdateien)

Element **<Target>**:

Ein Element **<Target>** definiert die Instanz einer Zielklasse, für die der Parameter gilt.

## Syntax

Übergeordnetes Element: **<Parameter>**

Es sind mehrere Vorkommen desselben Parameters aus denselben Katalogen zulässig, wenn sie für unterschiedliche Instanzen ihres Ziels gelten.

Die folgenden Attribute werden unterstützt:

*Tabelle 14. Attribute*

Attribut	Erforderlich	Typ	Beschreibung
<i>class</i>	Ja	string	Gibt den Namen der Zielklasse an.
<i>instance</i>	Nein*	string	Gibt den Namen der Instanz einer Klasse an.
<i>match</i>	Nein*	string	Gibt den regulären Ausdruck an, der auf die erkannten Instanznamen angewendet wird.

\* Es darf nur ein einziges der folgenden Attribute angegeben werden: *instance* oder *match*.

## Verwendung

Einige Parameter gelten nicht für das gesamte System, sondern nur für ein bestimmtes Objekt. Ein Beispiel ist das Verzeichnis *home* eines Benutzers, das im Profil *chuserProfile.xml* angegeben ist. Dieser Parameter gilt für einen bestimmten Benutzer (*root*, *guest*) in einem ladbaren Modul (*files*, *LDAP*). In diesem Beispiel sind *user* und *module* zwei Zielklassen. Der Parameter *home* gilt für spezielle Instanzen dieser Zielklassen. Die Instanz "guest" der Klasse "user" und die Datei "files" der Klasse "module".

Wenn die Attribute *class* und *instance* auf die leere Zeichenfolge gesetzt sind, wird eine Erkennung für diesen Parameter durchgeführt, wenn der Befehl **artexget** für ein solches Profil ausgeführt wird, die in der entsprechenden Katalogdatei deklarierte Erkennungsmethode ausgeführt wird und ein Parameter im Ausgabeprofil für jede erkannte Instanz des Parameters erstellt wird. Sehen Sie sich Beispiel 1 an.

Wenn die Attribute *class* und *instance* beide angegeben werden, ist das Ziel vollständig qualifiziert, und der Parameter gilt nur für die spezielle Instanz der Zielklasse. Sehen Sie sich Beispiel 2 an.

Wenn die Attribute *class* und *match* beide angegeben werden, wird wie zuvor eine Erkennung durchgeführt, aber es werden nur die Zielinstanzen erkannt, deren Namen dem regulären Ausdruck entsprechen, der im Attribut *match* angegeben ist. Sehen Sie sich Beispiel 3 an.

Beim Schreiben eines Beispielprofils müssen die Attribute *class* und *instance* leer bleiben. Wenn die leere Zielklasse gefunden wird, erkennt der Befehl **artexget** die Liste der Instanzen dieser Zielklasse (alle Benutzer oder Subsysteme auf dem System), bevor er die Werte abrufen.

Wenn Sie den Befehl **artexset** für eine nicht erkannte Zielklasse ausgeführt wird, wird eine Warnung angezeigt:

```
0590-216 Einige Parameter im Profil erfordern eine Zielerkennung und werden ignoriert.
```

## Beispiele

1. Ein Beispiel für ein Profil mit Zielen vor der Erkennung ist das Profil *chuserProfile.xml*, das das Ausgangsverzeichnis eines Benutzers definiert. Im Folgenden sehen Sie ein Beispielprofil:

```
<Profile version="2.0.0" origin="reference" readOnly="true">
 <Catalog id="chuserParam" version="2.0">
 <Parameter name="home">
 <Target class="" instance=""/>
 </Parameter>
 </Catalog>
</Profile>
```

2. Nach der Erkennung enthält das Profil *chuserProfile.xml* eine Kopie des Parameters "home" für jeden erkannten Benutzer in jedem der erkannten ladbaren Module:

```
<Profile version="2.0.0" origin="get">
 <Catalog id="chuserParam" version="2.0">

 <Parameter name="home" value="/">
 <Target class="user" instance="root"/>
 <Target class="module" instance="files"/>
 </Parameter>

 <Parameter name="home" value="/etc">
 <Target class="user" instance="daemon"/>
 <Target class="module" instance="files"/>
 </Parameter>

 ...

 </Catalog>
</Profile>
```

3. Das folgende Profil verwendet das Attribut *match*, um das Ausgangsverzeichnis aller Benutzer mit einem Namen, der mit *u* beginnt, im Modul "files" zu erkennen:

```
<Profile version="2.0.0" origin="reference" readOnly="true"
 <Catalog id="chuserParam" version="2.0">
 <Parameter name="home">
 <Target class="user" match="^u"/>
 <Target class="module" instance="files"/>
 </Parameter>
 </Catalog>
</Profile>
```

## Zugehörige Informationen

Element **<Discover>** (in Katalogdateien)

## Kataloge für AIX Runtime Expert schreiben

Katalogdateien werden vom Framework AIX Runtime Expert intern verwendet.

Die Katalogdateien enthalten die Parameterdefinitionen und Bindungsinformationen für die Konfigurationsmethoden, die die Befehle beschreiben, die zum Abrufen oder Setzen von Parameterwerten verwendet werden. Katalogdateien sind für das System, das optimiert und konfiguriert wird, lokal.

### Katalogkonzepte von AIX Runtime Expert:

Katalogdateien enthalten alle Informationen, die erforderlich sind, um Operationen für Parameter, einschließlich Definitionen, Nutzungsbedingungen und Konfigurationsmethoden, durchzuführen. Katalogdateien dürfen von Benutzern nicht direkt bearbeitet werden und werden nur von der AIX Runtime Expert-Kernengine verwendet.

Kataloge werden gleichzeitig mit der Kernengine von AIX Runtime Expert auf einem System installiert. Wenn neue Kataloge mit Komponenten oder Anwendungen anderer Anbieter, die auf einem System installiert sind, verlinkt werden, muss unbedingt sichergestellt werden, dass sie eine Version aufweisen, die mit der Kernengine von AIX Runtime Expert kompatibel ist.

#### *Katalogposition:*

Die Katalogdateien von AIX Runtime Expert sind im Verzeichnis `/etc/security/artex/catalogs` gespeichert.

Der Name einer Katalogdatei muss exakt mit dem Wert des zugehörigen Attributs *id* übereinstimmen, dem die Erweiterung `.xml` angefügt wird. Ein Katalogname mit dem Namen `commandParam.xml` muss beispielsweise den Wert `commandParam` für das Attribut *id* haben.

Damit der Katalog von dem Profil, der auf diesen Katalog verweist, gefunden werden kann, muss der Katalog in der XML-Datei des Katalogs und im Element **<Catalog>** der XML-Datei des Profils denselben Namen haben. Standardmäßig sucht die Kernengine von AIX Runtime Expert im Standardverzeichnis `/etc/security/artex/catalogs` nach Katalogen. Dieses Verhalten kann (ausschließlich für den Rootbenutzer) mit der Umgebungsvariablen `ARTEX_CATALOG_PATH` geändert werden. Mit dieser Umgebungsvariablen können mehrere Verzeichnisse angegeben werden, die dann mit dem Trennzeichen `:` voneinander getrennt werden müssen.

#### *Katalogprozess:*

Schritte zum Schreiben eines neuen Katalogs für AIX Runtime Expert.

Sie müssen die folgenden Schritte ausführen, um einen neuen Katalog für AIX Runtime Expert zu schreiben:

1. Erstellen Sie eine Liste der Parameter, die Sie in die Katalogdatei aufnehmen möchten.
2. Erstellen Sie für jeden Parameter ein Element **<ParameterDef>**.
3. Wenn mehrere Parameter denselben Befehl für eine Operation **<Get>**, **<Set>**, **<Discover>** oder **<Diff>** verwenden:
  - Definieren Sie ein Element **<CfgMethod>** oben im Katalog.
  - Verwenden Sie das Attribut *cfgmethod*, um Werte von der Konfigurationsmethode zu übernehmen.
4. Wenn mehrere Parameter derselben Bedingung unterliegen, definieren Sie ein Element **<ConstraintDef>** oben im Katalog.
5. Für jeden Parameter:
  - a. Definieren Sie die Operationen **<Get type="current">** und **<Get type="nextboot">** für jeden Parameter, entweder direkt unter dem Element **<ParameterDef>** oder referenziert unter dem Element **<CfgMethod>** oder einer beliebigen Kombination.
  - b. Definieren Sie alle unterstützten Operationen **<Set>** für jeden Parameter, entweder direkt unter dem Element **<ParameterDef>** oder unter dem referenzierten Element **<CfgMethod>** oder mit einer Kombination dieser Möglichkeiten.
  - c. Wenn der Parameter ein Ziel erfordert:
    - 1) Definieren Sie die unterstützten Zielklassen mit dem Attribut *targetClass*.
    - 2) Definieren Sie die Operation "discover", entweder direkt unter dem Element **<ParameterDef>** oder unter dem referenzierten Element **<CfgMethod>** oder mit einer beliebigen Kombination dieser Möglichkeiten. In den meisten Fällen wird die Methode "discover" in einer Konfigurationsmethode definiert.
  - d. Wenn der Parameter einen Warmstart erfordert, damit eine Änderung wirksam wird, fügen Sie das Attribut *reboot=true* hinzu.
  - e. Wenn der Parameter einer Bedingung unterliegt, definieren Sie ein Element **<ConstraintDef>** unter dem Element **<ParameterDef>**, oder verwenden Sie das Attribut "constraint", um auf eine vorhandene Bedingung zu verweisen.
6. Gehen Sie zum Testen der Katalogdatei wie folgt vor:
  - a. Erstellen Sie ein Profil mit allen Parametern, die in der Katalogdatei definiert sind.
  - b. Verwenden Sie den Befehl **artexget -r**, um Werte zu erfassen und die Operationen **<Discover>** und **<Get>** zu testen.
  - c. Verwenden Sie den Befehl **artexset -c -F -R -l all** im resultierenden Profil, um die Operationen **<Set>** und **<Diff>** zu testen.
  - d. Außerdem können Sie diesen beiden Befehlen die Flags **-g 3 -g COMMANDS** hinzufügen, um weitere Informationen zur generierten Befehlszeile abzurufen, um die angeforderte Operation auszuführen.

## Zugehörige Informationen

Stammelement **<Catalog>**

### Katalogelemente in AIX Runtime Expert:

*Element <Catalog>*:

Das Element **<Catalog>** ist das Stammelement für alle Katalogdateien.

### Syntax

Die folgenden Attribute werden unterstützt:

Tabelle 15. Attribute

Attribut	Erforderlich	Typ	Beschreibung
<i>id</i>	Ja	string	Gibt den Namen des Katalogs an. Dieser Name muss auf dem System eindeutig sein.
<i>version</i>	Nein	string	Gibt die Versionsnummer des Katalogs an.
<i>date</i>	Nein	dateTime	Gibt das Erstellungsdatum an. Das Format ist <i>YYYY-MM-TTThh:mm:ss</i> .
<i>priority</i>	Nein	integer	Gibt die Ausführungsreihenfolge des Katalogs relativ zu anderen in den set-Methoden an. Der Standardwert ist 0.
<i>inherit</i>	Nein	string	Gibt den Namen des zu übernehmenden Katalogs an.

Die folgenden untergeordneten Elemente werden unterstützt. Die Spalte *Anzahl* definiert, wie viele Vorkommen des untergeordneten Elements zulässig sind:

Tabelle 16. Untergeordnete Elemente

Untergeordnetes Element	Erforderlich	Anzahl	Beschreibung
<ShortDescription>	Nein	0-1	Die Kurzbeschreibung für den Katalog.
<Description>	Nein	0-1	Die Langbeschreibung für den Katalog.
<SubCat>	Nein	0-n	Die Unterkategorie.
<ParameterDef>	Nein	0-n	Enthält die Eigenschaften eines Parameters.
<ConstraintDef>	Nein	0-n	Die Integritätsbedingungsdefinition für den Parameter (Bedingungen und Unterbrechungsbefehle).
<CfgMethod>	Nein	0-n	Die Definition der Konfigurationsmethode.
<PrereqDef>	Nein	0-n	Definiert eine Voraussetzung.
<PropertyDef>	Nein	0-n	Definiert eine Eigenschaft.
<SeedDef>	Nein	0-n	Definiert einen Seed.

## Attribute

Tabelle 17. Attribute

Attribut	Beschreibung
<i>id</i>	Das Attribut <i>id</i> muss dem Katalogdateinamen ohne die Erweiterung <i>.xml</i> entsprechen. Die Katalog-ID wird in Profilen mit dem Element <Catalog> referenziert.
<i>priority</i>	Das Attribut <i>priority</i> wird verwendet wenn die set-Methoden eines bestimmten Katalogs vor oder nach den set-Methoden anderer Kataloge ausgeführt werden müssen, wenn sie in demselben Profil enthalten sind (z. B. im Verbundprofil <b>default.xml</b> ). Die Standardpriorität eines Katalogs ist 0.  Wenn zwei Kataloge dieselbe Priorität haben, gilt die Regel, dass die set-Methoden dieser Kataloge in nicht definierter Reihenfolge ausgeführt werden. Wenn ein Katalog eine Priorität mit einem positiven Wert hat, werden die set-Methoden dieses Katalogs vor den anderen in absteigender Reihenfolge ihrer Priorität ausgeführt. Wenn ein Katalog eine Priorität mit einem negativen Wert hat, werden die set-Methoden dieses Katalogs nach den anderen in absteigender Reihenfolge ihrer Priorität ausgeführt.

Tabelle 17. Attribute (Forts.)

Attribut	Beschreibung
<i>Version</i>	Das Attribut <i>Version</i> ist in beiden Profilen und Katalogen vorhanden. Mithilfe der Version kann festgestellt werden, ob Profile und Kataloge mit der Kernengine von AIX Runtime Expert und miteinander kompatibel sind. Weitere Einzelheiten finden Sie in der Beschreibung des Attributs <i>Version</i> .
<i>Date</i>	Das Attribut <i>date</i> wird momentan für das Element <b>&lt;Catalog&gt;</b> nicht verwendet. Es wird für künftige Verwendung und Wartbarkeit bereitgestellt.
<i>inherit</i>	Das Attribut <i>inherit</i> gibt den Namen eines Katalogs ohne die Erweiterung .xml an, aus dem Elemente übernommen werden. Alle im übernommenen Katalog definierten Elemente sind im Hauptkatalog verfügbar, als wären sie lokal definiert worden.

## Beispiel

Im Folgenden sehen Sie ein Beispiel für einen Katalog mit dem Attribut *priority*. Der Katalog `aixpertParam.xml` definiert Sicherheitsoptionen und muss definiert werden, nachdem alle anderen Kataloge definiert wurden. Deshalb wird die Priorität dieses Katalogs auf einen hohen negativen Wert gesetzt.

```
<Catalog id="aixpertParam" version="2.0" priority="-1000">
```

## Zugehörige Informationen

Element `<ConstraintDef>`

Element `<CfgMethod>`

Elemente `<Description>` und `<ShortDescription>`

Element `<ParameterDef>`

Element `<SubCat>`

*Versionsattribut:*

## Syntax

Die Version eines Katalogs wird als Attribut im Format *MM.mmm* geschrieben, wobei *MM* für die Hauptversionsnummer (Major Number) und *mmm* für die Nebenversionsnummer (Minor Number) steht.

```
<Catalog id="commandParam" version="2.0">
```

## Hauptversionsnummer

Die Hauptversionsnummer ist für alle Kataloge von AIX Runtime Expert, die auf einem System installiert sind, und an allen Positionen im gesamten Framework AIX Runtime Expert, an denen diese Versionsnummer referenziert wird, dieselbe. Diese Hauptversionsnummer wird mit jeder Hauptänderung des XML-Schemas der Profile und Kataloge um eins erhöht.

Setzen Sie beim Erstellen eines neuen Katalogs die Hauptversionsnummer auf die aktuelle Versionsnummer der Kernengine von AIX Runtime Expert. Sie finden diese Versionsnummer in den Standardkatalogdateien, die mit der Dateigruppe `artex.base.rte` bereitgestellt werden.

Wenn ein Befehl **artexget** für ein Profil aufgerufen wird, deren Hauptversionsnummer von der Versionsnummer abweicht, die in der Kernengine von AIX Runtime Expert referenziert wird, schlägt der Befehl mit dem folgenden Fehler fehl:

```
0590-117 Versionsfehler
```

```
Dieses Profil wurde in einer von ARTEX nicht unterstützten Version erstellt.
```

Außerdem wird im Hinblick auf die Kompatibilität empfohlen, dass ein Profil und ein Katalog diese Hauptversionsnummer haben. Ein Profil referenziert Kataloge mit einer bestimmten Versionsnummer. Wenn die Hauptversionsnummer des Profils nicht mit der Hauptversionsnummer des Katalogs übereinstimmt, zeigt jeder Befehl von AIX Runtime Expert eine Warnung an, um den Benutzer auf nicht vorhersehbare Ereignisse hinzuweisen:

0590-218 Die Katalogversion weicht von der im Profil referenzierten Version ab.

### Nebenversionsnummer

Die Nebenversionsnummer ist für jeden Katalog spezifisch und wird jedes Mal erhöht, wenn eine wichtige Änderung dazu führt, dass der Katalog mit der vorherigen Version nicht mehr kompatibel ist. Ein Profil referenziert Kataloge mit einer bestimmten Versionsnummer. Wenn die Nebenversionsnummer des Profils nicht mit der Nebenversionsnummer des Katalogs übereinstimmt, gibt jeder Befehl von AIX Runtime Expert eine Warnung aus, um den Benutzer auf nicht vorhersehbare Ergebnisse hinzuweisen:

0590-218 Die Katalogversion weicht von der im Profil referenzierten Version ab.

Wenn Sie ein neues Beispielprofil oder einen neuen Katalog erstellen, setzen Sie die Nebenversionsnummer auf 0.

*Elemente <Description> und <ShortDescription>:*

Beschreibungen sind optionale Informationstextfelder, die verschiedenen Elementen in den Katalogdateien hinzugefügt werden können. Diese Felder sind optional, aber es wird empfohlen, dass Katalogautoren sie verwenden, um das übergeordnete Element zu dokumentieren.

### Syntax

Ein Element **<ShortDescription>** kann eines der folgenden übergeordneten Elemente haben:

- **<Catalog>**
- **<SubCat>**

Ein Element **<Description>** kann eines der folgenden übergeordneten Elemente haben:

- **<Catalog>**
- **<SubCat>**
- **<ParameterDef>**
- **<ConstraintDef>**

Der Inhalt der Elemente **<Description>** und **<ShortDescription>** ist entweder eine einfache Zeichenfolge oder eine übersetzte Nachricht, die von einem der folgenden Elemente definiert wird: **<NLSCatalog>**, **<NLSSmitHelp>** oder **<NLSCCommand>**. Weitere Informationen finden Sie unter Globalisierungsunterstützung.

### Verwendung

Momentan wird bei der Ausführung des Befehls **artexget** mit dem Flag **-i** nur die Beschreibung von **<ParameterDef>**-Elementen abgerufen und angezeigt. Es wird empfohlen, die Globalisierung für den Text in diesen Beschreibungsfeldern bereitzustellen.

Die Beschreibungsfelder der anderen Elemente werden momentan nicht vom Framework AIX Runtime Expert verwendet, aber sie sollten für künftige Verwendung und für Dokumentationszwecke bereitgestellt werden.

### Beispiel

1. Im Folgenden sehen Sie ein Beispiel für Beschreibungsfelder:



```

<ShortDescription>
 chuser parameters
</ShortDescription>
<Description>
 Parameter definition for the chuser command
</Description>

```

2. Im Folgenden sehen Sie dasselbe Beispiel mit übersetzten Nachrichten aus der Nachrichtendatei `artexcat.cat`:

```

<ShortDescription>
<NLSCatalog catalog="artexcat.cat" setNum="12" msgNum="1">
 chuser parameters
</NLSCatalog>
</ShortDescription>
<Description>
<NLSCatalog catalog="artexcat.cat" setNum="12" msgNum="2">
 Parameter definition for the chuser command
</NLSCatalog>
</Description>

```

## Zugehörige Informationen

Globalisierungsunterstützung

*Globalisierungsunterstützung:*

In diesem Abschnitt wird beschrieben, wie die Globalisierung in den beschreibenden Feldern der Kataloge von AIX Runtime Expert implementiert wird.

## Syntax

Übergeordnetes Element: **<Description>**, **<ShortDescription>**

Das übergeordnete Element darf nur ein einziges der folgenden untergeordneten Elemente enthalten:

*Tabelle 18. Untergeordnete Elemente*

Untergeordnetes Element	Erforderlich	Nummer	Beschreibung
<b>&lt;NLSCatalog&gt;</b>	Nein	0-1	Die in einem Nachrichtenkatalog enthaltene Zeichenfolge.
<b>&lt;NLSSmitHelp&gt;</b>	Nein	0-1	Die in einer SMIT-HTML-Hilfedatei enthaltene Zeichenfolge.
<b>&lt;NLSCommand&gt;</b>	Nein	0-1	Die von einem AIX-Befehl ausgegebene Zeichenfolge.

## Sprachenunterstützungskatalog

Das Globalisierungsformat für den Sprachenunterstützungskatalog wird verwendet, wenn die anzuzeigende lokalisierte Nachricht in einem vorhandenen Nachrichtenkatalog im Format `catgets()` enthalten ist.

Das Element **<NLSCatalog>** enthält die folgenden Attribute:

Tabelle 19. Attribute

Attribut	Erforderlich	Typ	Beschreibung
<i>catalog</i>	Ja	string	Der Name des Katalogs, in dem die Nachricht enthalten ist.
<i>setNum</i>	Ja	integer	Die Nummer des Nachrichtensatzes, in dem die Nachricht enthalten ist.
<i>msgNum</i>	Ja	integer	Die Nummer der Nachricht im Nachrichtensatz.

Wenn der lokalisierte Nachrichtenkatalog nicht vorhanden ist, wird stattdessen die Standardnachricht angezeigt. Die Standardnachricht wird optional als Inhalt des Elements **<NLSCatalog>** eingeschlossen. Es wird empfohlen, eine Standardnachricht bereitzustellen.

### SMIT-Hilfe für die Sprachenunterstützung

Das Globalisierungsformat für die Sprachenunterstützung der SMIT-Hilfe wird verwendet, wenn die anzuzeigende lokalisierte Nachricht bereits in einer SMIT-HTML-Hilfedatei enthalten ist.

Das Element **<NLSSmitHelp>** enthält das folgende Attribut:

Tabelle 20. Attribute

Attribut	Erforderlich	Typ	Beschreibung
<i>msgId</i>	Ja	integer	Das Feld <b>help_msg_id</b> , das in der SMIT-Zeilengruppe enthalten ist.

Wenn die lokalisierte Hilfedatei nicht vorhanden ist, wird stattdessen die Standardnachricht angezeigt. Die Standardnachricht wird optional als Inhalt des Elements **<NLSSmitHelp>** eingeschlossen. Es wird empfohlen, eine Standardnachricht bereitzustellen.

### Sprachenunterstützungsbefehl

Das Globalisierungsformat für Sprachenunterstützungsbefehle wird verwendet, wenn die anzuzeigende lokalisierte Nachricht von einem AIX-Befehl ausgegeben wird. Dies gilt für alle Optimierungsbefehle (wie **no**, **vmo**), die ein Flag **-h** enthalten, damit der Hilfetext für einen bestimmten Parameter angezeigt wird.

Das Element **<NLSCCommand>** enthält das folgende Attribut:

Tabelle 21. Attribut

Attribut	Erforderlich	Typ	Beschreibung
<i>command</i>	Befehl	string	Der auszuführende Shellausdruck.

### Beispiele

1. Beispiel für das Element **<NLSCatalog>** aus dem Katalog `chssysParam.xml` von AIX Runtime Expert, einschließlich einer Standardnachricht:

```
<Description>
 <NLSCatalog catalog="artexcat.cat" setNum="10" msgNum="2">
 Ändert eine Subsystemdefinition in der Objektklasse subsystem. </NLSCatalog>
</Description>
```

2. Beispiel für das Element **<NLSSmitHelp>**:

```
<Description>
 <NLSSmithHelp msgId="055136"/>
</Description>
```

3. Beispiel für das Element **<NLSSCommand>** aus dem Katalog schedoParam.xml:

```
<Description>
 <NLSSCommand command="/usr/sbin/schedo -h maxspin | /usr/bin/tail -n +2"/>
</Description>
```

Element **<SubCat>**:

Unterkategorien, optionale Parameter und Subsets in einem Katalog können mit dem Element **<SubCat>** in einer Katalogdatei angegeben werden.

## Syntax

Übergeordnetes Element: **<Catalog>**, **<SubCat>**

Die folgenden Attribute werden unterstützt:

Tabelle 22. Attribute

Attribut	Erforderlich	Typ	Beschreibung
<i>id</i>	Ja	string	Gibt den Namen der Katalogunterkategorie an. Dieser Name muss in einer Katalogdatei eindeutig sein.

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 23. Untergeordnete Elemente

Untergeordnetes Element	Erforderlich	Beschreibung
<b>&lt;ShortDescription&gt;</b>	Nein	Die Kurzbeschreibung der Unterkategorie.
<b>&lt;Description&gt;</b>	Nein	Die lange Beschreibung der Unterkategorie.
<b>&lt;SubCat&gt;</b>	Nein	Die verschachtelte Unterkategorie. Dieses Element kann mehrfach vorkommen.
<b>&lt;ParameterDef&gt;</b>	Nein	Enthält die Eigenschaften eines Parameters. Dieses Element kann mehrfach vorkommen.

## Attribut

Eine Unterkategorie ist für einen Katalog lokal:

- Eine Unterkategorie-ID ist in einer Katalogdatei eindeutig.
- Mehrere Kataloge können dieselbe Unterkategorie-ID verwenden.

Die in einem Katalog definierten Unterkategorien müssen den Unterkategorien, die im zugehörigen Beispielfprofil berichtet werden, exakt entsprechen.

## Zugehörige Informationen

Elemente **<Description>** und **<ShortDescription>**

Element **<SubCat>**

Element **<ParameterDef>**

Element `<ParameterDef>`:

AIX Runtime Expert wird in einer Katalogdatei mit dem Element `<ParameterDef>` definiert.

## Syntax

Übergeordnetes Element: `<Catalog>`, `<ParameterDef>`

Die folgenden Attribute werden unterstützt:

Tabelle 24. Attribute

Attribut	Erforderlich	Typ	Beschreibung
<i>name</i>	Ja	string	Gibt den Namen des Parameters an. Dieser Name muss für jeden Katalog eindeutig sein.
<i>type</i>	Ja	string	Gibt den Parametertyp aus Sicht der Kernengine an.
<i>targetClass</i>	Nein	string	Gibt die Zielklassen für den Parameter an, sofern vorhanden.
<i>reboot</i>	Nein	boolean	Wenn dieses Attribut auf "true" gesetzt wird, ist ein Warmstart erforderlich. Der Standardwert ist "false".
<i>cfgmethod</i>	Nein	string	Gibt die ID der auf <code>&lt;Katalogebene&gt;</code> definierten Konfigurationsmethode an, die die Methoden enthält, die für diesen Parameter verwendet werden sollen.
<i>constraint</i>	Nein	string	Gibt die ID einer Bedingung an, die auf Ebene des Elements <code>&lt;Catalog&gt;</code> für die aktuelle Katalogdatei definiert ist.
<i>priority</i>	Nein	integer	Ausführungsrang dieses Parameters in der set-Methode relativ zu anderen Parametern in diesem Katalog. Der Standardwert ist 0.

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 25. Untergeordnete Elemente

Untergeordnetes Element	Erforderlich	Beschreibung
<code>&lt;Description&gt;</code>	Nein	Die Beschreibung des Parameters.
<code>&lt;ConstraintDef&gt;</code>	Nein	Die Integritätsbedingungsdefinition für den Parameter (Unterbrechungsbefehle).
<code>&lt;Get&gt;</code>	Nein	Die Konfigurationsmethodendefinition für die get-Operation. Dieses Element kann mehrfach vorkommen.
<code>&lt;Set&gt;</code>	Nein	Die Konfigurationsmethodendefinition für die set-Operation. Dieses Element kann mehrfach vorkommen.
<code>&lt;Diff&gt;</code>	no	Die Konfigurationsmethodendefinition für die diff-Operation.
<code>&lt;Discover&gt;</code>	no	Die Konfigurationsmethodendefinition für die Zielerkennung.

## Attribute

Tabelle 26. Attribute

Attribut	Beschreibung
<i>name</i>	Das Attribut "name" identifiziert einen Parameter in einer Katalogdatei eindeutig. Weitere Informationen finden Sie im Abschnitt zum Parameterattribut "name".
<i>type</i>	Das erforderliche Attribut "type" gibt den Wertetyp des Parameters an. Die unterstützten Werte sind im Folgenden aufgelistet: <ul style="list-style-type: none"> <li>• string für alphanumerische Zeichenfolgen</li> <li>• integer für numerische Werte</li> <li>• integer-bi für numerische Werte mit einem optionalen Suffix K, M, G, T, P oder E (Großbuchstabe oder Kleinbuchstabe) für "kilo", "mega", "giga", "tera", "peta" bzw. "exa". Diese Suffixe werden als Potenz von 1024 interpretiert.</li> <li>• integer-si für numerische Werte mit dem optionalen Suffix SI. Entspricht abgesehen davon, dass die Suffixe als Potenzen von 1000 interpretiert werden, dem Typ "integer-bi".</li> <li>• boolean für boolesche Werte. Die unterstützten Werte sind 0 und 1.</li> <li>• binary für Binärwerte, codiert als Base-64-Zeichenfolgen in Profilen.</li> </ul>
<i>reboot</i>	Der Standardwert für das boolesche Attribut "reboot" ist "false". Wenn eine Parameteränderung einen Warmstart erfordert, muss das Attribut "reboot" des Parameters auf "true" gesetzt werden.  AIX Runtime Expert selbst führt keinen Warmstart von Systemen durch. Standardmäßig erzwingt der Befehl <b>artexset</b> die Einstellung des Parameters "reboot" nicht. Wenn das Profil den Parameter "reboot" enthält, schlägt der Befehl fehl:  0590-502: Das Profil hat Parameter, die einen Warmstart erfordern. Das Profil wurde nicht definiert. Verwenden Sie das Flag -l, um set für alle Parameter zu erzwingen.  Wenn der Befehl <b>artexset</b> mit dem entsprechenden Flag <b>-l</b> aufgerufen wird, setzt der Befehl den Wert und gibt eine Warnung aus, die die Benutzer darauf hinweist, dass ein Warmstart erforderlich ist, damit die Änderungen wirksam werden:  0590-206 Es ist eine manuelle nachbereitende Operation erforderlich, damit die Änderungen wirksam werden. Führen Sie einen Warmstart des Systems durch.
<i>priority</i>	Standardmäßig werden Parameter in nicht definierter Reihenfolge vom Befehl <b>artexset</b> gesetzt. Das Attribut <i>priority</i> kann verwendet werden, um dieses Verhalten zu ändern und die Festlegung eines Parameters vor oder nach anderen Parametern zu erzwingen.  Die Standardpriorität ist 0. Das Attribut "priority" kann verwendet werden, um diese Standardpriorität in einen beliebigen ganzzahligen Wert zwischen -2147483648 und 2147483647 zu ändern. Parameter mit einer höheren Priorität werden vor Parametern mit einer niedrigeren Priorität ausgeführt. Die Reihenfolge, in der Parameter mit derselben Priorität ausgeführt werden, ist nicht definiert.
<i>targetClass</i>	Einigen Parametern muss ein Ziel zugeordnet werden. Diesbezügliche Informationen finden Sie im Abschnitt zum Element "target" eines Profils. Für diese Parameter muss das Attribut <i>targetClass</i> auf eine durch Kommas getrennte Liste der unterstützten Zielklassen gesetzt werden.

Tabelle 26. Attribute (Forts.)

Attribut	Beschreibung
<i>cfgmethod</i>	Ein Element <b>&lt;ParameterDef&gt;</b> kann Befehlszeilenelemente von einem Element <b>&lt;CfgMethod&gt;</b> übernehmen, indem das Attribut <i>id</i> dieser Konfigurationsmethode mit dem Attribut <i>cfgmethod</i> referenziert wird. Weitere Informationen zu Konfigurationsmethoden finden Sie im Abschnitt zum Element <b>&lt;CfgMethod&gt;</b> .
<i>constraint</i>	Ein Element <b>&lt;ParameterDef&gt;</b> kann das Attribut "constraint" verwenden, um das Attribut "id" eines Elements <b>&lt;ConstraintDef&gt;</b> zu referenzieren und damit darauf hinzuweisen, dass der Parameter der Bedingung unterliegt. Weitere Informationen zu Integritätsbedingungen finden Sie im Abschnitt zum Element <b>&lt;ConstraintDef&gt;</b> .

## Beispiele

1. Im Folgenden sehen Sie ein Beispiel für eine Parameterdefinition mit einem alternativen ganzzahligen Typ: *Größe des Kernel-Heapspeichers* aus der Katalogdatei vmoParam.xml:

```
<ParameterDef name="kernel_heap_psize" type="integer-bi">
```

Wenn der Wert dieses Parameters mit einem Befehl **artexget** extrahiert wird, gleicht das Ergebnis dem folgenden Beispiel (Auszug aus dem resultierenden Profil).

```
<Parameter name="kernel_heap_psize" value="16M"... />
```

Der Parameterwert wird je nach Typ anders interpretiert:

- Da der Wert mit dem Typ "integer-bi" deklariert ist, ist der Wert 16M = 16.777.216.
- Wenn "integer-si" als Typ deklariert wäre, wäre der Wert "16M"=16.000.000.

2. Beispiel für einen Binärparameter: Anerkannte Signaturdatenbank tsd.dat im Katalog tsdParam.xml:

```
<ParameterDef name="tsdatabse" type="binary">
```

3. Beispiel für einen Parameter mit einem Attribut *reboot*. Der Typ des Parameters "dump" im Katalog sysdumpdevParam.xml:

```
<ParameterDef name="type_of_dump" type="string" reboot="true">
```

4. Beispiel für einen Parameter mit einer einzigen Zielklasse: Der Parameter *addr* aus dem Katalog mktcpipParam.xml gilt für eine bestimmte Netzchnittstelle:

```
<ParameterDef name="addr" type="string" cfgmethod="mktcpip" targetClass="interface">
```

5. Beispiel für einen Parameter mit mehreren Zielklassen: Der Parameter "namingspecification" aus der Datei coreParam.xml gibt für einen bestimmten Benutzer (root, admin, guest usw.) in einer bestimmten Registry (files, LDAP).

```
<ParameterDef name="namingspecification" type="string" reboot="true" targetClass="user,registry"
cfgmethod="coremgt">
```

6. Beispiel für die Verwendung des Attributs *cfgmethod*: Bei der Operation **<Get type="current">** übernimmt der Parameter "fixed" des Katalogs chlicenseParam.xml das Element **<Command>** der Konfigurationsmethode *chlicense*, definiert aber auch eigene lokale Elemente **<Filter>** und **<Mask>** für diese Operation:

```
<CfgMethod id="chlicense">
 <Get type="current">
 <Command>lslicense -c -A</Command>
 </Get>
</CfgMethod>
<ParameterDef name="fixed" cfgmethod="chlicense" type="integer">
 <Get type="current">
 <Filter>tail -n 1 | cut -d: -f3</Filter>
 <Mask value="1">(.*)</Mask>
 </Get>
</ParameterDef>
```

7. Beispiel für die Verwendung des Attributs "constraint": Der Parameter "authorizations" des Katalogs authParam.xml unterliegt der Bedingung **setkst**, die zuvor in einem Element **<ConstraintDef>** definiert wurde:

```
<ParameterDef name="authorizations" cfgmethod="cat" constraint="setkst" type="string">
```

### Zugehörige Informationen

Attribut *name*

Attribut *name*:

Der Name eines Parameters wird häufig durch den Befehl vorgegeben, der zum Abrufen oder Setzen des Parameters verwendet wird.

Parameternamen müssen in einer Katalogdatei eindeutig sein. Die Eindeutigkeit ist erforderlich, um sicherzustellen, dass dem Element **<Parameter>** in einem Profil ein eindeutiges Element **<ParameterDef>** in einer Katalogdatei zugeordnet werden kann.

- Wenn der Befehl **get** mehrere Parameter-Wert-Paare anzeigt, kann das Element **<Mask>** verwendet werden, um mehrere Parameter aus einer einzigen Befehlsausgabe zu extrahieren. Dies ist nur möglich, wenn der Name des Parameters mit dem Namen übereinstimmt, der in der Ausgabe des Befehls **get** verwendet wird.
- Wenn der Befehl **set** mehrere Parameter-Wert-Paare akzeptiert, können die %n- und %v1-Sequenzen in einem Element **<Argument>** verwendet werden, um mehrere Parameter mit einem einzigen Befehl zu setzen. Dies ist nur möglich, wenn der Name des Parameters mit dem Namen übereinstimmt, der vom Befehl **set** verwendet wird.

### Beispiele

1. Der Befehl **raso -a**, der im Katalog rasoParam.xml verwendet wird, zeigt einen Parameter pro Anzeigzeile an:

```
kern_heap_noexec = 0
kernel_noexec = 1
mbuf_heap_noexec = 0
mtrc_commonbufsize = 485
```

In diesem einfachen Fall sind die Parameternamen *kernel\_heap\_noexec*, *kernel\_noexec* usw.

2. Der Befehl, der in der Konfigurationsmethode **get** des Katalogs acctctlParam.xml verwendet wird, zeigt ein Ergebnis an, das schwieriger zu parsen ist. Der Name des Parameters ist nicht nur in einen nicht formatierten Satz integriert, sondern es sind auch beide Parameternamen und Parameterwerte lokalisiert. Die Konfigurationsmethoden "get" müssen den Befehl ausführen, gleichzeitig die Umgebungsvariable **LANG=C** setzen und in jeder Zeile die Schlüsselwörter durch relevante Parameternamen ersetzen:

```
Advanced Accounting ist nicht aktiv.
E-Mail-Benachrichtigung ist inaktiviert.
Die zu verwendende aktuelle E-Mail-Adresse ist nicht definiert.
Wiederherstellung der CPU-Abrechnungszeit im Turbomodus ist aktiviert (false).
```

Im vorherigen Beispiel sind die ausgewählten Variablennamen *accounting*, *email*, *email\_addr* und *turacct*.

### Zugehörige Informationen

- Element **<Parameter>**
- Element **<Mask>**
- Erweiterung der Befehlszeilenelemente

Element `<ConstraintDef>`:

## Syntax

Übergeordnetes Element: `<Catalog>`, `<ParameterDef>`

Die folgenden Attribute werden unterstützt:

Table 27. Attribute

Attribut	Erforderlich	Typ	Beschreibung
<code>id</code>	Nein*	string	Gibt den Namen der Parameterbedingung an.

\*Dieses Attribut muss für Elemente `<Constraint>` angegeben werden, die auf Katalogebene definiert werden.

Die folgenden untergeordneten Elemente werden unterstützt:

Table 28. Untergeordnete Elemente

Untergeordnete Elemente	Erforderlich	Beschreibung
<code>&lt;Description&gt;</code>	Nein	Die Beschreibung des Unterbrechungsbefehls.
<code>&lt;PreOp&gt;</code>	Nein	Die Unterbrechungsoperationen, die vor dem Setzen des Parameterwerts ausgeführt werden sollen.
<code>&lt;PostOp&gt;</code>	Nein	Die Unterbrechungsoperationen, die nach dem Setzen des Parameterwerts ausgeführt werden sollen.
<code>&lt;BuiltIn&gt;</code>	Nein	Die integrierte Unterbrechungsoperation. Dieses Element kann mehrfach vorkommen.

## Verwendung

Einige Optimierungs- und Konfigurationsparameter können Unterbrechungsoperationen erfordern, damit die Änderungen wirksam werden. Eine Unterbrechungsoperation ist eine Operation, die den Zugriff auf einen Service oder eine Einheit vorübergehend unterbrechen können. Typische Unterbrechungsoperationen sind der Neustart eines Dämons, das Abhängen oder Anhängen eines Dateisystems, das Aktivieren und das Inaktivieren einer Netzadapterkarte. Das Programm AIX Runtime Expert verwendet Bedingungen, um anzuzeigen, dass ein Parameter eine Unterbrechungsoperation erfordert, damit Änderungen wirksam werden. Ein Element `<ConstraintDef>` wird verwendet, um eine solche Bedingung zu definieren.

Eine Bedingung kann wie folgt definiert werden:

- Sie können die Bedingung in einem Element `<ParameterDef>` definieren, wenn die Bedingung nur für diesen einen Parameter gilt.
- Auf Katalogebene muss das Element `<ConstraintDef>` ein Attribut `id` haben, damit die Bedingung später in Elementen `<ParameterDef>` referenziert werden kann.

## Integrierte Bedingung

Das Element `<BuiltIn>` enthält kein Attribut und kein untergeordnetes Element.

Die integrierte Bedingung definiert Operationen, die in der Kernengine fest codiert sind. Es ist momentan nur eine einzige integrierte Bedingung definiert: `bosboot`. Integrierte Bedingungen unterscheiden sich inso-



fern von anderen Unterbrechungsoperationen, dass der Befehl *bosboot* von AIX Runtime Expert nie ausgeführt wird. Die Kernengine gibt nur eine Warnung aus, die darauf hinweist, dass *bosboot* erforderlich ist, damit die Änderungen wirksam werden.

0590-206 Es ist eine manuelle nachbereitende Operation erforderlich, damit die Änderungen wirksam werden. Führen Sie *bosboot* aus.

## PreOp- und PostOp-Bedingungen

Das Element **<PreOp>** definiert verbindliche Befehle (Shellausdrücke), die ausgeführt werden sollen, bevor der Parameterwert von der Konfigurationsmethode "set" gesetzt wird. Das Element **<PostOp>** definiert verbindliche Befehle, die nach der Ausführung der Konfigurationsmethode "set" ausgeführt werden sollen.

Ein Element **<ConstraintDef>** muss 0 oder ein untergeordnetes Element **<PreOp>** und 0 oder ein untergeordnetes Element **<PostOp>** enthalten.

## Beispiele

1. Beispiel für eine integrierte Bedingung (auf Katalogebene)

```
<ConstraintDef id="bosboot">
 <Description>
 <NLSCatalog catalog="artexcat.cat" setNum="51" msgNum="3">
 bosboot
 </NLSCatalog>
 </Description>
 <Built>Inbosboot</BuiltIn>
</ConstraintDef>
```

2. Beispiel für eine Bedingung **<PreOp>**: Die Bedingung *clic* im Katalog *trustchkParam.xml*. Beachten Sie, dass der Befehl **preop** in diesem Beispiel keine Operation ausführt, sondern nur prüft, ob die vom Befehl **set** benötigte Kernelerweiterung vorhanden ist. Wenn die Kernelerweiterung nicht installiert ist, schlägt die im Element **<PreOp>** definierte Bedingung fehl, und der Befehl **set** wird nicht ausgeführt:

```
<ConstraintDef id="clic">
 <Description>
 <NLSCatalog catalog="artexcat.cat" setNum="48" msgNum="3">
 Check that the clic.rte kernel extension is installed.
 </NLSCatalog>
 </Description>
 <PreOp>plslpp -l "clic*"</PreOp>
</ConstraintDef>
```

3. Beispiel für eine Bedingung **<PostOp>**: Die Bedingung "set Kernel Security Tables" im Katalog *authParam.xml*. Die geänderten Datenbanken müssen nur ein einziges Mal in den Kernel geladen werden, nachdem alle Änderungen vorgenommen wurden.

```
<ConstraintDef id="setkst">
 <Description>
 <NLSCatalog catalog="artexcat.cat" setNum="5" msgNum="3">
 Send the authorizations database to the KST (Kernel Security Tables)
 </NLSCatalog></Description>
 <PostOp>/usr/sbin/setkst -t auth >/dev/null</PostOp>
</ConstraintDef>
```

Element **<CfgMethod>**:

## Syntax

Übergeordnetes Element: **<Catalog>**

Das folgende Attribut wird unterstützt:

Tabelle 29. Attribut

Attribut	Erforderlich	Typ	Beschreibung
<i>id</i>	Ja	string	Gibt den Namen der Konfigurationsmethode an.

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 30. Untergeordnete Elemente

Untergeordnete Elemente	Erforderlich	Nummer	Beschreibung
<Get>	Nein	0-1	Die Konfigurationsmethodendefinition für die get-Operation. Dieses Element kann mehrfach vorkommen.
<Set>	Nein	0-1	Die Konfigurationsmethodendefinition für die set-Operation. Dieses Element kann mehrfach vorkommen.
<Diff>	Nein	0-1	Die Konfigurationsmethodendefinition für die diff-Operation.
<Discover>	Nein	0-1	Die Konfigurationsmethodendefinition für die Zielerkennung.
<Property>	Nein	0-n	Ordnet den Parametern mit der Konfigurationsmethode eine Eigenschaft zu.

## Verwendung

Das Element <CfgMethod> definiert eine Konfigurationsmethode, die später mit einem Parameter unter Verwendung des Attributs *cfgmethod* des Elements <ParameterDef> referenziert werden kann. Daraufhin übernimmt der Parameter alle Elemente, die unter der referenzierten Konfigurationsmethode definiert sind.

Ja nach Parameter bietet eine Konfiguration mehrere Vorteile gegenüber einer lokalen Definition:

- Sie vereinfacht die Katalogdatei, weil die Duplizierung derselben Befehlszeilenelemente für mehrere Parameter vermieden wird.
- Sie ermöglicht, dass mehrere Parameter als ein einziger Befehl behandelt werden können.

## Beispiel

Der Katalog *vmoParam.xml* definiert eine Vielzahl von Parametern, die alle dieselbe Konfigurationsmethode verwenden. Im Folgenden sehen Sie eine vereinfachte Version dieses Katalogs:

```
<Catalog id="vmoParam" version="2.1">
 <CfgMethod id="vmo">
 <Get type="current">
 <Command>/usr/sbin/vmo -a</Command>
 <Mask name="1" value="2">[:space:]]*(.*) = (.*)</Mask>
 </Get>

 <Get type="nextboot">
 <Command>/usr/sbin/vmo -r -a</Command>
 <Mask name="1" value="2">[:space:]]*(.*) = (.*)</Mask>
 </Get>
 </CfgMethod>
</Catalog>
```

```

 <Set type="permanent">
 <Command>/usr/sbin/vmo -p%a</Command>
 <Argument>%n=%v1</Argument>
 </Set>

 <Set type="nextboot">
 <Command>/usr/sbin/vmo -r%a</Command>
 <Argument>%n=%v1</Argument>
 </Set>
 </CfgMethod>

 <ParameterDef name="ame_maxfree_mem" cfgmethod="vmo" type="integer" />
 <ParameterDef name="ame_min_ucpool_size" cfgmethod="vmo" type="integer" />
 <ParameterDef name="ame_minfree_mem" cfgmethod="vmo" type="integer" />
 ...
</Catalog>

```

### Zugehörige Informationen

- Befehlszeilengenerierung
- Element **<Get>**
- Element **<Set>**

Element **<Get>**:

### Syntax

Übergeordnetes Element: **<CfgMethod>**, **<ParameterDef>**

Das folgende Attribut wird unterstützt:

Tabelle 31. Attribut

Attribut	Erforderlich	Typ	Beschreibung
<i>type</i>	Ja	string	Gibt den Typ des Befehls <b>get</b> an (current oder <i>nextboot</i> ).

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 32. Untergeordnete Elemente

Untergeordnete Elemente	Erforderlich	Nummer	Beschreibung
<b>&lt;Command&gt;</b>	Nein	0-1	Befehl
<b>&lt;Argument&gt;</b>	Nein	0-1	Befehlszeilenargumente
<b>&lt;Stdin&gt;</b>	Nein	0-1	Vom Element <b>&lt;Stdin&gt;</b> unterstützte Argumente
<b>&lt;Filter&gt;</b>	Nein	0-1	Filter
<b>&lt;Mask&gt;</b>	Nein	0-1	Maske für Erfassung der Ausgabe
<b>&lt;Prereq&gt;</b>	Nein	0-n	Weist der Get-Operation eine Voraussetzung zu.

Das Element **<Command>** muss für jeden Parameter definiert werden, entweder auf der Ebene von **<CfgMethod>** oder direkt auf der Ebene von **<ParameterDef>**.

### Verwendung

Das Element **<Get>** beschreibt, wie der Wert eines bestimmten Parameters erfasst wird. Es kann entweder direkt unter dem Element **<ParameterDef>** oder unter einem Element **<CfgMethod>**, das im Element **<ParameterDef>** mit dem Attribut *cfgmethod* referenziert wird, oder mit einer Kombination dieser zwei Optionen verwendet werden.

Für jeden Parameter müssen zwei Get-Elemente definiert werden, eines für jeden unterstützten Wert des Attributs *type*:

- Get **type="current"** gibt die Methode an, die verwendet wird, um den Laufzeitwert des Parameters abzurufen.
- Get **type="nextboot"** gibt die Methode an, die ausgeführt wird, um den Wert abzurufen, den der Parameter nach dem nächsten Warmstart des Systems annimmt.
- Die auszuführende Methode "get" richtet sich nach der ausgeführten Operation:
  - Wenn der Befehl **artexget** mit dem Flag *-r* aufgerufen wird, wird die get-Methode "current" verwendet.
  - Wenn der Befehl **artexget** mit dem Flag *-n* aufgerufen wird, wird die get-Methode *nextboot* verwendet.
  - Wenn der Befehl **artexget** mit dem Flag *-p* aufgerufen wird, richtet sich die ausgeführte Methode nach den Parametern, die mit dem Attribut *applyType* eingegeben werden. Die get-Methode "current" wird für die Parameter verwendet, deren Attribut *applyType* auf "runtime" gesetzt ist, und die get-Methode *nextboot* wird für die Parameter verwendet, deren Attribut *applyType* auf "reboot" gesetzt ist.

## Zugehörige Informationen

Befehlszeilengenerierung

Element **<Mask>**

Element **<Set>**:

Das Element **<Set>** definiert, wie eine Befehlszeile zum Setzen des Werts eines Parameters erstellt wird.

## Syntax

Übergeordnetes Element: **<CfgMethod>**, **<ParameterDef>**

Das folgende Attribut wird unterstützt:

Tabelle 33. Attribut

Attribut	Erforderlich	Typ	Beschreibung
<i>type</i>	Ja	string	Gibt den Typ des Befehls <b>set</b> an (current oder nextboot).

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 34. Untergeordnete Elemente

Untergeordnete Elemente	Erforderlich	Nummer	Beschreibung
<b>&lt;Command&gt;</b>	Nein	0-1	Befehl
<b>&lt;Argument&gt;</b>	Nein	0-1	Befehlszeilenargumente
<b>&lt;Stdin&gt;</b>	Nein	0-1	Stdin-Argumente
<b>&lt;Prereq&gt;</b>	Nein	0-n	Weist der <b>&lt;Set&gt;</b> -Operation eine Voraussetzung zu.

**Anmerkung:** Das Element **<Command>** muss für jeden Parameter definiert werden, entweder auf der Ebene von **<CfgMethod>** oder direkt auf der Ebene von **<ParameterDef>**.

## Verwendung

Es gibt drei Typen von **<Set>**-Elementen, die für jeden Parameter definiert werden können und mit dem erforderlichen Attribut *type* angegeben werden:

- Set **type="current"** definiert eine set-Operation, die den Wert des Parameters nur für die aktuelle Sitzung ändert. Alle Änderungen, die mit der set-Operation vorgenommen werden, gehen nach einem Warmstart des Systems verloren.
- Set **type="nextboot"** definiert eine set-Operation, die den Wert des Parameters erst nach dem nächsten Warmstart des Systems ändert. Der aktuelle Wert wird nicht geändert.
- Set **type="permanent"** definiert eine set-Operation, die den current- (aktuell) und den nextboot-Wert (nächster Warmstart) des Parameters ändert.

Der Typ der ausgeführten set-Operation wird auf der Basis der Parameter, die bei der Ausführung des Befehls **artexset** angegeben werden, basierend auf dem Attribut *applyType* des Parameters im Profil festgelegt. In der folgenden Tabelle sind die set-Methoden zusammengefasst, die in Abhängigkeit von den set-Methoden, die in der Katalogdatei definiert sind, abhängig vom Attribut *applyType* für den Parameter ausgeführt werden:

Tabelle 35. Set-Methoden - Definierte Typen für Set-Methoden und Parameterattribut *applyType*

current	nextboot	permanent	runtime	nextboot
0	0	0	nicht gesetzt (Fehler)	nicht gesetzt (Fehler)
0	0	1	set permanent	nicht gesetzt (Fehler)
0	1	0	set nextboot + Warnung	set nextboot
0	1	1	set permanent	nicht gesetzt (Fehler)
1	0	0	set current + Warnung	set nextboot
1	0	1	set permanent	nicht gesetzt (Fehler)
1	1	0	set current set nextboot	set nextboot
1	1	1	set permanent	set nextboot

## Zugehörige Informationen

### Befehlszeilengenerierung

Element **<Diff>**:

Das Element **<Diff>** definiert, wie eine Befehlszeile für den Vergleich von zwei Werten eines Parameters erstellt wird.

### Syntax

Übergeordnetes Element: **<CfgMethod>**, **<ParameterDef>**

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 36. Untergeordnete Elemente

Untergeordnete Elemente	Erforderlich	Beschreibung
<b>&lt;Command&gt;</b>	Nein	Befehl
<b>&lt;Argument&gt;</b>	Nein	Befehlszeilenargumente
<b>&lt;Stdin&gt;</b>	Nein	Stdin-Argumente
<b>&lt;Filter&gt;</b>	Nein	Filter
<b>&lt;Mask&gt;</b>	Nein	Maske für Erfassung der Ausgabe

**Anmerkung:** Das Element `<Command>` muss für jeden Parameter definiert werden, entweder auf der Ebene von `<CfgMethod>` oder direkt auf der Ebene von `<ParameterDef>`.

### Verwendung

Das Element `<Diff>` ist gewöhnlich nicht erforderlich, da das Framework weiß, wie zwei Parameterwerte intern auf der Basis des Typs (string, integer, integer-bi, binary usw.) verglichen werden. Falls der interne Vergleich jedoch nicht für einen bestimmten Parameter angepasst ist, kann stattdessen auch ein externer Befehl verwendet werden.

### Beispiel

Das folgende Element `<Diff>` kann für die meisten Parameter verwendet werden, aber die interne Vergleichsfunktion ist effizienter. Das Element `<Diff>` verwendet den Befehl `diff`, um zwei Dateien zu vergleichen, die die beiden Werte enthalten:

```
<Diff>
 <Command>/usr/bin/diff %f1 %f2; echo $?</Command>
</Diff>
```

### Zugehörige Informationen

Befehlszeilengenerierung

Element `<Mask>`

*Element <Discover>:*

Das Element `<Discover>` definiert, wie eine Befehlszeile zur Erkennung von Zielen für einen Parameter erstellt wird, der sie unterstützt.

### Syntax

Übergeordnetes Element: `<CfgMethod>`, `<ParameterDef>`

Die folgenden untergeordneten Elemente werden unterstützt:

*Tabelle 37. Untergeordnete Elemente*

Untergeordnete Elemente	Erforderlich	Nummer	Beschreibung
<code>&lt;Command&gt;</code>	Nein	0-1	Befehl
<code>&lt;Prereq&gt;</code>	Nein	0-n	Weist der Discover-Operation eine Voraussetzung zu.

**Anmerkung:** Das Element `<Command>` muss für jeden Parameter definiert werden, entweder auf der Ebene von `<CfgMethod>` oder direkt auf der Ebene von `<ParameterDef>`.

### Verwendung

Ein Befehl "discover" wird verwendet, um die Liste der Zielinstanzen für einen bestimmten Parameter abzurufen.

Die Ausgabe eines Befehls "discover" für einen Parameter, der N Zielklassen unterstützt, hat das folgende Format:

```

class_1=inst_1_1;class_2=inst_2_1;...;class_N=inst_N_1
class_1=inst_1_2;class_2=inst_2_2;...;
class_N=inst_N_2class_1=inst_1_3;
class_2=inst_2_3;...;class_N=inst_N_3
...

```

Der Befehl **artexget** generiert und führt einen Befehl "discover" für Parameter aus, die eines der folgenden Kriterien erfüllen:

- Der Parameter enthält ein Element **<Target>** mit leeren Attributen *class* und *instance*. **<Target class="" instance="" />**
- Der Parameter enthält mindestens ein Element **<Target>** mit einem Attribut *match*: **<Target class="..." match="..." />**

Der Befehl **artexset** erfordert zusätzlich, dass die folgenden beiden Kriterien erfüllt sind:

- Der Befehl **artexset** wird mit dem Flag *-d* aufgerufen.
- Im Element **<Parameter>** des Profils ist das Attribut *setDiscover* auf "true" gesetzt.

## Beispiele

1. Der Katalog `mktcpipParam.xml` verwendet den folgenden Befehl "discover", um die Liste der Netz-schnittstellen abzurufen, die auf dem System definiert sind:

```

<Discover>
 <Command>
 /usr/sbin/lsdev -C -c if -F "name" | /usr/bin/sed -e 's/^/interface=/'
 </Command>
</Discover>

```

Dieser Befehl gibt die folgende Ausgabe zurück:

```

interface=en0
interface=et0
interface=lo0

```

2. Der Katalog `chuserParam.xml` verwendet den folgenden Befehl **discover**, um die Liste aller Benutzer für alle ladbaren Authentifizierungsmodule abzurufen:

```

<Discover>
 <Command>
 /usr/sbin/lsuser -a registry ALL | /usr/bin/sed -e "s/\(.*\) registry=\(.*\) /module=\2;user=\1/g"
 </Command>
</Discover>

```

Dieser Befehl gibt die folgende Ausgabe zurück:

```

module=LDAP;user=daemon
module=LDAP;user=bin
module=LDAP;user=sys
module=LDAP;user=adm
...
module=files;user=root
module=files;user=daemon
module=files;user=bin
module=files;user=sys
module=files;user=adm
...

```

Element **<Command>**:

Das Element **<Command>** definiert den Basisbefehl, der verwendet wird, um die vom übergeordneten Element definierte Operation auszuführen.

## Syntax

Übergeordnetes Element: `<Get>`, `<Set>`, `<Diff>`, `<Discover>`, `<PrereqDef>`, `<Prereq>`, `<PropertyDef>`, `<Property>`, `<Command>`

## Verwendung

Der Inhalt des Elements `<Command>` wird gemäß der Beschreibung im Abschnitt Erweiterung von Befehlszeilenelementen erweitert und mit anderen Befehlszeilenelementen zu einer vollständigen Befehlszeile kombiniert. Weitere Einzelheiten finden Sie im Abschnitt Befehlszeilengenerierung.

Einige Zeichen, die in Shellausdrücken häufig verwendet werden, wie z. B. `<`, `>` und `&`, sind in XML-Dokumenten nicht zulässig. Diese Zeichen müssen durch die entsprechende XML-Entität ersetzt werden:

Table 38. XML-Entitäten

Zeichen	XML-Entität
<code>&lt;</code>	<code>&amp;lt;</code>
<code>&gt;</code>	<code>&amp;gt;</code>
<code>&amp;</code>	<code>&amp;amp;</code>

Alternativ kann ein CDATA-Abschnitt verwendet werden, wenn der Ausdruck viele solcher Zeichen enthält. Die CDATA-Abschnitte beginnen mit `<![CDATA[` und enden mit `]]>`.

Das Element `<Command>` muss für jede unterstützte Operation jedes Parameters definiert werden, entweder auf der Ebene von `<CfgMethod>` oder auf der Ebene von `<ParameterDef>`.

## Beispiel

Der Katalog `envParam.xml` definiert einen Parameter mit dem Namen "profile", der den Inhalt der Datei `/etc/profile` darstellt. Für diesen Parameter verwendet das Element `<Get>` den Befehl `cat`, um den Inhalt der Datei `/etc/profile` zu erfassen:

```
<ParameterDef name="profile">
 <Get type="current">
 <Command>/usr/bin/cat /etc/environment</Command>
 </Get>
</ParameterDef>
```

## Zugehörige Informationen

Befehlszeilengenerierung

Erweiterung der Befehlszeilenelemente

Element `<Argument>`:

## Syntax

Übergeordnetes Element: `<Get>`, `<Set>`, `<Diff>`, `<PrereqDef>`, `<Prereq>`, `<PropertyDef>`, `<Property>`

## Verwendung

Der Inhalt des Elements `<Argument>` wird gemäß der Beschreibung im Abschnitt Erweiterung von Befehlszeilenelementen erweitert und mit dem Element `<Command>` oder `<Stdin>` zu einer vollständigen Befehlszeile kombiniert. Weitere Einzelheiten finden Sie im Abschnitt Befehlszeilengenerierung.



Einige Zeichen, die in Shellausdrücken häufig verwendet werden, wie z. B. <, > und &, sind in XML-Dokumenten nicht zulässig. Diese Zeichen müssen durch die entsprechende XML-Entität ersetzt werden:

Tabelle 39. XML-Entitäten

Zeichen	XML-Entität
<	&lt;
>	&gt;
&	&amp;

Alternativ kann ein CDATA-Abschnitt verwendet werden, wenn der Ausdruck viele solcher Zeichen enthält. Die CDATA-Abschnitte beginnen mit `<![CDATA[` und enden mit `]]>`.

### Beispiel

Der Katalog `vmoParam.xml` verwendet das Element `<Argument>`, um dem Befehl `vmo` für jeden Parameter `vmo` im Profil ein Argument hinzuzufügen.

```
<CfgMethod id="vmo">
 <Set type="permanent">
 <Command>/usr/sbin/vmo -p%a</Command>
 <Argument> -o %n=%v1</Argument>
 </Set>
</CfgMethod>
```

### Zugehörige Informationen

Befehlszeilengenerierung

Erweiterung der Befehlszeilenelemente

Element `<Stdin>`:

### Syntax

Übergeordnetes Element: `<Get>`, `<Set>`, `<Diff>`, `<PrereqDef>`, `<Prereq>`, `<PropertyDef>`, `<Property>`

### Verwendung

Der Inhalt des Elements `<Stdin>` wird entsprechend der Beschreibung im Abschnitt Erweiterung von Befehlszeilenelementen erweitert, und die Ergebnisdaten werden in die Standardeingabe der Befehlszeile geschrieben, die für die im übergeordneten Element definierte Operation generiert wurde.

### Beispiel

Der Katalog `envParam.xml` definiert einen Parameter, mit dem Namen "profile", der den Inhalt der Datei `/etc/profile` darstellt. Für diesen Parameter schreibt die `set`-Operation den Wert des Parameters in die Standardeingabe des Befehls `cat`, um die Datei `/etc/profile` zu überschreiben:

```
<ParameterDef name="profile">
 <Set type="permanent">
 <Command>/usr/bin/cat > /etc/profile</Command>
 <Stdin>%v1</Stdin>
 </Set>
</Get>
```

## Zugehörige Informationen

Befehlszeilengenerierung

Erweiterung der Befehlszeilenelemente

Element `<Filter>`:

### Syntax

Übergeordnetes Element: `<Get>`, `<Diff>`, `<PropertyDef>`, `<Property>`

### Verwendung

Der Inhalt des Elements `<Filter>` ist ein Befehl, an den die Ausgabe der Befehlszeile, die für die im übergeordneten Element definierte Operation generiert wurde, als Eingabe übergeben wird.

Einige Zeichen, die in Shellausdrücken häufig verwendet werden, wie z. B. `<`, `>` und `&`, sind in XML-Dokumenten nicht zulässig. Diese Zeichen müssen durch die entsprechende XML-Entität ersetzt werden:

Tabelle 40. XML-Entitäten

Zeichen	XML-Entität
<code>&lt;</code>	<code>&amp;lt;</code>
<code>&gt;</code>	<code>&amp;gt;</code>
<code>&amp;</code>	<code>&amp;amp;</code>

Alternativ kann ein CDATA-Abschnitt verwendet werden, wenn der Ausdruck viele solcher Zeichen enthält. Die CDATA-Abschnitte beginnen mit `<![CDATA[` und enden mit `]]>`.

### Beispiel

Der Katalog `nfsParam.xml` verwendet das Element `<Filter>` für die Operation "get" des Parameters `v4_root_node`, um den Stammknoten aus der Ausgabe des Befehls `nfsd -getnode` zu extrahieren:

```
<ParameterDef id="v4_root_node">
 <Get type="current">
 <Command>
 /usr/sbin/nfsd -getnodes
 </Command>
 <Filter>
 /usr/bin/awk -F: 'NR == 2 { printf("%s", $1) }'
 </Filter>
 </Get>
</ParameterDef>
```

## Zugehörige Informationen

Befehlszeilengenerierung

Element `<Mask>`:

### Syntax

Übergeordnetes Element: `<Get>`, `<Diff>`, `<Discover>` (nur unter `<SeedDef>`), `<PropertyDef>`, `<Property>`

Die folgenden Attribute werden bei Verwendung in einem Element `<Get>` oder `<Diff>` unterstützt:

Tabelle 41. Attribute

Attribut	Erforderlich	Typ	Beschreibung
<i>name</i>	Nein	integer	Gibt den Index des Unterausdrucks an, der dem Namen des Parameters entspricht. Die gültigen Werte sind 1 und 2.
<i>value</i>	Nein	integer	Gibt den Index des Unterausdrucks an, der dem Wert des Parameters entspricht. Die gültigen Werte sind 1 und 2.

Die folgenden Attribute werden bei Verwendung im Unterelement **<Discover>** eines Elements **<Seed-Def>** unterstützt:

Tabelle 42. Attribute

Attribut	Erforderlich	Typ	Beschreibung
<i>catalog</i>	Ja	integer	Gibt den Index des Unterausdrucks an, der dem Namen des Katalogs entspricht. Die gültigen Werte sind 1, 2 und 3.
<i>name</i>	Ja	integer	Gibt den Index des Unterausdrucks an, der dem Namen des Parameters entspricht. Die gültigen Werte sind 1, 2 und 3.
<i>target</i>	Nein	integer	Gibt den Index des Unterausdrucks an, der dem Namen des Ziels entspricht. Die gültigen Werte sind 1, 2 und 3.

Das folgende Attribut wird bei Verwendung in einem Element **<PropertyDef>** oder **<Property>** unterstützt:

Tabelle 43. Attribut

Attribut	Erforderlich	Typ	Beschreibung
<i>value</i>	Nein	integer	Gibt den Index des Unterausdrucks an, der dem Namen des Parameters entspricht. Dieses Attribut muss auf "1" gesetzt werden, wenn es definiert wird.

## Verwendung

Das Element **<Mask>** definiert einen regulären Ausdruck, der auf jede Zeile der Befehlsausgabe angewendet wird, um Daten aus diesen Zeilen zu extrahieren. Welche Daten extrahiert werden, richtet sich danach, wo das Element **<Mask>** verwendet wird.

Wenn keine Attribute angegeben werden, wird die letzte Zeile in der Befehlsausgabe, die dem regulären Ausdruck entspricht, zum Extrahieren der Daten verwendet. Die extrahierten Daten sind der Teil der Zeile, der dem regulären Ausdruck entspricht. Bei Verwendung in einem Element **<Get>** oder **<Diff>** werden die extrahierten Daten als Parameterwert verwendet. Bei Verwendung in einem Element **<Property-Def>** oder **<Property>** werden die extrahierten Daten als Eigenschaftswert verwendet.

Wenn nur das Attribut *value* angegeben wird, muss es auf 1 gesetzt werden, und der reguläre Ausdruck darf nur einen einzigen Unterausdruck enthalten. Die letzte Zeile in der Befehlsausgabe, die dem regulären Ausdruck entspricht, wird zum Extrahieren der Daten verwendet. Die extrahierten Daten sind der Teil der Zeile, der dem ersten (und einzigen) Unterausdruck entspricht. Bei Verwendung in einem Element **<Get>** oder **<Diff>** werden die extrahierten Daten als Parameterwert verwendet. Bei Verwendung in einem Element **<PropertyDef>** oder **<Property>** werden die extrahierten Daten als Eigenschaftswert verwendet.

Wenn die Attribute *name* und *value* angegeben werden, muss eines dieser Attribute auf 1 und das andere auf 2 gesetzt werden, und der reguläre Ausdruck muss zwei Unterausdrücke enthalten. Aus jeder Zeile der Befehlsausgabe, die dem regulären Ausdruck entspricht, werden ein *Name* und ein *Wert* extrahiert. Bei Verwendung in einem Element **<Get>** wird der Name als Parametername und der Wert als Parameterwert verwendet. Bei Verwendung in einem Element **<Diff>** wird der Name als Parametername und der Wert als Vergleichsergebnis verwendet. Wenn diese Funktion verwendet wird, können die Werte mehrerer Parameter mithilfe eines einzigen Befehls **get** extrahiert und mehrere Parameter mit einem einzigen Befehl **diff** verglichen werden.

Bei Verwendung im Unterelement **<Discover>** eines Elements **<SeedDef>** müssen die Attribute "catalog" und "name" angegeben werden. Aus jeder Zeile der Befehlsausgabe, die dem regulären Ausdruck entspricht, werden ein Katalogname und ein Parametername extrahiert. Wenn ein Katalog, der dem extrahierten Katalognamen entspricht, auf dem System gefunden wird und dieser eine Definition für einen Parameter enthält, der dem extrahierten Parameternamen entspricht, wird ein Parameter in das Profil eingefügt. Das optionale Zielargument kann hinzugefügt werden, um eine Zieldefinition für jeden erkannten Parameter zu extrahieren. Die Zieldefinition muss das Format einer durch Semikolons getrennten Liste mit Klasse=Instanz-Paaren haben, z. B. Klasse1=Instanz1;Klasse2=Instanz2;....

## Beispiele

1. Der Katalog vmoParam.xml verwendet das Element **<Mask>** mit den Attributen *name* und *value*, um alle Parameterwerte aus einem einzigen Befehl **vmo -a** zu extrahieren:

```
<CfgMethod id="vmo">
 <Get type="current">
 <Command>/usr/sbin/vmo -a</Command>
 <Mask name="1" value="2">[[[:space:]]*(.*) = (.*)</Mask>
 </Get>
</CfgMethod>
```

2. Wenn der Katalog vmoParam.xml so geschrieben wird, dass ein gesonderter Befehl zum Erfassen des Werts jedes Parameters verwendet wird, kann das Element **<Mask>** mit gesetztem Attribut *value* und ohne Attribut *name* verwendet werden:

```
<CfgMethod id="vmo">
 <Get type="current">
 <Command>/usr/sbin/vmo -o %n</Command>
 <Mask value="1"> = (.*)</Mask>
 </Get>
</CfgMethod>
```

3. Alternative mit einem regulären Ausdruck, der nur dem Wert entspricht:

```
<CfgMethod id="vmo">
 <Get type="current">
 <Command>/usr/sbin/vmo -o %n</Command>
 <Mask>[^]*${</Mask>
 </Get>
</CfgMethod>
```

Von den drei vorherigen Beispielen ist das erste das effizienteste, da nur ein einziger Befehl erforderlich ist, um alle Parameter des Befehls **vmo** zu erfassen. Die Beispiele 2 und 3 generieren einen separaten Befehl für jeden Parameter des Befehls **vmo**, da der Parametername im Element **<Command>** verwendet wird.

4. Das folgende Element **<SeedDef>** definiert einen Seed, der für die Erkennung aller Attribute aller Einheiten verwendet werden kann. Das Element verwendet ein Ziel, um die Einheit anzugeben, für die die Operation ausgeführt werden soll:

```
<SeedDef name="devAttr">
 <Discover>
 <Command>
 /usr/sbin/lsdev -F 'name class subclass type' |
 while read DEV CLASS SUBCLASS TYPE
 do
 /usr/sbin/lsattr -F attribute -l $DEV |
 while read PAR
 do
 echo device=$DEV devParam.$CLASS.$SUBCLASS.$TYPE $PAR
 done
 done
 </Command>
 <Mask target="1" catalog="2" name="3">(.) (.) (.) <Mark>
 </Discover>
</SeedDef>
```

Der Erkennungsbefehl gibt jedes erkannte Einheitenattribut im folgenden Format in einer separaten Zeile aus:

```
device=DeviceName devParam.Class.Subclass.Type AttributeName
```

Verwenden Sie beispielsweise den folgenden Befehl:

```
device=en0 devParam.if.EN.en tcp_recvspace
device=en0 devParam.if.EN.en tcp_sendspace
device=ent0 devParam.adapter.vdevice.IBM,1-lan alt_addr
device=ent0 devParam.adapter.vdevice.IBM,1-lan chksum_offload
```

## Zugehörige Informationen

Befehlszeilengenerierung

Element **<SeedDef>**:

Das Element **<SeedDef>** definiert einen Seed, der in einem Profil mithilfe eines Elements **<Seed>** verwendet werden kann.

### Syntax

Übergeordnetes Element: **<Catalog>**

Das folgende Attribut wird unterstützt:

*Tabelle 44. Attribut*

Attribut	Erforderlich	Typ	Beschreibung
<i>name</i>	Ja	string	Gibt den Namen des Seeds an. Dieser Name muss für jeden Katalog eindeutig sein.

Das folgende untergeordnete Element wird unterstützt:

Tabelle 45. Untergeordnetes Element

Untergeordnetes Element	Erforderlich	Beschreibung
<Discover>	Ja	Gibt den Befehl an, der zum Erkennen von Parametern verwendet wird.

## Verwendung

Seeds werden verwendet, um Parameter während einer Get-Operation dynamisch zu erkennen.

Wenn der Befehl **artexget** ausgegeben wird, wird jedes Element <Seed> im Eingabeprofil basierend auf den im entsprechenden Element <SeedDef> der Katalogdatei definierten Regeln zu einem oder mehreren Elementen <Parameter> erweitert. Dieser Prozess wird als Parametererkennung bezeichnet. Der Befehl **artexget** fährt anschließend wie gewöhnlich mit der Verarbeitung des erweiterten Profils fort.

Das Element <SeedDef> enthält nur ein einziges Unterelement <Discover>, das einen auszuführenden Befehl definiert, und eine Maske für die Extraktion von Parameternamen, Katalognamen (in Form von durch Doppelpunkten getrennten Listen ohne die Erweiterung *.xml*) und optional Zielen aus der Ausgabe des Befehls (im Format *Klasse1=Instanz1;Klasse2=Instanz2;...*). Für jede Zeile der Ausgabe wird der erste Katalog aus der durch Doppelpunkten getrennten Liste geladen, der auf dem System gefunden wird. Wenn eine Parameterdefinition in diesem Katalog gefunden wird, wird ein Parameter im Ausgabeprofil erstellt, der die Ziele enthält, die aus der Zeile extrahiert wurden. Zeilen aus der Befehlsausgabe, die der Maske nicht entsprechen, oder für die keine Katalogdatei gefunden wird oder die keine Parameterdefinition haben, obwohl sie in der Katalogdatei vorhanden sind, werden ignoriert.

## Beispiele

1. Der folgende Katalog definiert ein Element <SeedDef> mit dem Namen *vmoTunables*, das alle nicht eingeschränkten *vmoTunables*-Seeds erkennt, die von AIX Runtime Expert unterstützt werden:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Catalog id="vmoSeed">
 <SeedDef name="vmoTunables">
 <Discover>
 <Command>/usr/sbin/vmo -x | /usr/bin/awk -F, '{ print "vmoParam:" $1 }'</Command>
 <Mask catalog="1" name="2">(.*):(.*)/Mask</Mask>
 </Discover>
 </SeedDef>
</Catalog>
```

Der Erkennungsbefehl gibt jeden optimierbaren Parameter mit dem Namen des Katalogs, der die optimierbaren Parameter definiert, in einer separaten Zeile aus:

```
...
vmoParam:enhanced_affinity_vmpool_limit
vmoParam:esid_allocator
vmoParam:force_realias_lite
vmoParam:kernel_heap_size
...
```

Das folgende Profil verwendet den *vmoTunables*-Seed für die Erfassung aller nicht eingeschränkten *vmoTunables*-Seeds, die von AIX Runtime Expert unterstützt werden:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Profile>
 <Catalog id="vmoSeed">
 <Seed name="vmoTunables"/>
 </Catalog>
</Profile>
```

Wenn der Befehl **artexget -r** für ein Profil ausgeführt wird, generiert der Befehl ein Profil, das dem folgenden Beispiel gleicht:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Profile>
 <Catalog id="vmoParam">
```

```

...
<Parameter name="enhanced_affinity_vmpool_limit" value="10"/>
<Parameter name="esid_allocator" value="0"/>
<Parameter name="force_realias_lite" value="0"/>
<Parameter name="kernel_heap_psize" value="65536" applyType="nextboot" reboot="true"/>
...
</Catalog>
</Profile>

```

- Das folgende Element **<SeedDef>** definiert einen Seed, der für die Erkennung aller Attribute aller Einheiten verwendet wird. Das Element verwendet einen Zielseed, um die Einheit anzugeben, für die die Operation ausgeführt werden soll:

```

<SeedDef name="devAttr">
 <Discover>
 <Command>
 /usr/sbin/lsdev -F 'name class subclass type' |
 while read DEV CLASS SUBCLASS TYPE
 do
 /usr/sbin/lsattr -F attribute -l $DEV |
 while read PAR
 do
 echo device=$DEV devParam.$CLASS.$SUBCLASS.$TYPE:devParam.$CLASS
 .$SUBCLASS:devParam.$CLASS $PAR
 done
 done
 </Command>
 <Mask target="1" catalog="2" name="3">(.*) (.*) (.*)</Mask>
 </Discover>
</SeedDef>

```

Der Erkennungsbefehl gibt jedes erkannte Einheitenattribut im folgenden Format in einer separaten Zeile aus:

```

device=DeviceName devParam.Class.Subclass.Type:devParam.Class.Subclass:devParam.Class
AttributeName

```

Beispiel:

```

device=en0 devParam.if.EN.en:devParam.if.EN:devParam.if tcp_recvspace
device=en0 devParam.if.EN.en:devParam.if.EN:devParam.if tcp_sendspace
device=ent0 devParam.adapter.vdevice.IBM,l-lan:devParam.adapter.vdevice:devParam.adapter
alt_addr
device=ent0 devParam.adapter.vdevice.IBM,l-lan:devParam.adapter.vdevice:devParam.adapter
chksum_offload

```

*Element <Prereq>:*

Das Element **<Prereq>** weist **<Get>**-, **<Set>**- und **<Discover>**-Operationen eine Voraussetzung zu.

## Syntax

Übergeordnetes Element: **<Get>**, **<Set>** und **<Discover>**

Das folgende Attribut wird unterstützt:

Tabelle 46. Attribut

Attribut	Erforderlich	Typ	Beschreibung
<i>id</i>	Nein	string	Gibt eine eindeutige ID an.

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 47. Untergeordnete Elemente

Untergeordnetes Element	Erforderlich	Beschreibung
<Command>	Nein	Befehl
<Argument>	Nein	Befehlszeilenargumente
<Stdin>	Nein	Vom Element <Stdin> unterstützte Argumente
<ErrorMessage>	Nein	Auszugebende Nachricht, wenn Voraussetzungen nicht erfüllt sind

**Anmerkung:** Das Element <Command> muss für jede Voraussetzung auf <ParameterDef>-Ebene, auf <CfgMethod>-Ebene oder in einem Element <PrereqDef> definiert werden.

## Verwendung

**Prereq**-Elemente sind Befehle, die Bedingungen für die Verarbeitung einer <Get>-, <Set>- und <Discover>-Operation für Parameter festlegen, die diese <Get>-, <Set>- und <Discover>-Operationen verwenden. Parameter, für die ein **prereq**-Befehl scheitert (Rückgabecode ungleich null), werden ignoriert, und die in der Voraussetzung definierte Fehlermeldung wird angezeigt.

Das Element <Prereq> weist der übergeordneten <Get>-, <Set>- oder <Discover>-Operation eine Voraussetzung zu. Die Voraussetzung wird entweder lokal im Element <Prereq> definiert oder wird von einem Element <Prereq> oder <PrereqDef> höherer Ebene übernommen, das ein entsprechendes Attribut *id* hat.

Die Voraussetzungen eines Parameters werden alle lokal im Element <ParameterDef> definiert. Außerdem werden die Eigenschaften der Voraussetzung in der Konfigurationmethode des Parameters definiert, wenn eine Konfigurationmethode verwendet wird. Wenn eine Voraussetzung in einem Element <CfgMethod> definiert wird, haben deshalb automatisch alle Elemente <ParameterDef>, die die Konfigurationmethode verwenden, automatisch diese Voraussetzung (obwohl einige Elemente die Voraussetzung möglicherweise lokal neu definieren).

Die Elemente <Command>, <Argument>, <Stdin> und <ErrorMessage>, die eine Voraussetzung für eine bestimmte Operation definieren, werden in der folgenden Reihenfolge gesucht:

- Im Unterelement <Prereq> der relevanten Operation des Elements <ParameterDef>
- Wenn das Element <ParameterDef> ein Attribut *cfgmethod* hat, im Unterelement <Prereq>, das ein entsprechendes Attribut *id* der relevanten Operation der Konfigurationmethode hat
- Im Element <PrereqDef> des Katalogs, das eine entsprechende ID hat

## Beispiel

Das folgende Beispiel definiert eine Voraussetzung, die prüft, ob die Parameter **netaddr** und **netaddr6** auf das System angewendet werden, auf dem sie erfasst werden:

```
<ParameterDef name="netaddr" type="string" targetClass="device" cfgmethod="attr">
 <Set type="permanent">
 <Prereq>
 <Command>[[` /usr/bin/uname -f` = %p[nodeId]]]</Command>
 <ErrorMessage>Parameter cannot be applied to a different node</ErrorMessage>
 </Prereq>
 </Set>
</ParameterDef>
```



```

<ParameterDef name="netaddr6" type="string" targetClass="device" cfgmethod="attr">
 <Set type="permanent">
 <Prereq>
 <Command>[[` /usr/bin/uname -f` = %p[nodeId]]]</Command>
 <ErrorMessage>Parameter cannot be applied to a different node</ErrorMessage>
 </Prereq>
 </Set>
</ParameterDef>

```

In diesem Beispiel wird der Test zweimal ausgeführt: einmal für den Parameter **netaddr** und einmal für den Parameter **netaddr6**. Die Verarbeitung wird zweimal ausgeführt, weil jeder Parameter eine eigene Voraussetzung mit einem eigenen Element **<Command>** hat. Im Abschnitt „Element **<PrereqDef>**“ finden Sie ein Beispiel, in dem der Test nur ein einziges Mal ausgeführt werden muss.

### Zugehörige Informationen

- „Befehlszeilengenerierung“ auf Seite 125
- „Element **<PrereqDef>**“

Element **<PrereqDef>**:

Das Element **<PrereqDef>**, das später in einem Element **<Prereq>** verwendet werden kann.

### Syntax

Übergeordnetes Element: **<Catalog>**

Das folgende Attribut wird unterstützt:

Table 48. Attribut

Attribut	Erforderlich	Typ	Beschreibung
<i>name</i>	Ja	string	Gibt den Namen der Eigenschaft an.

Die folgenden untergeordneten Elemente werden unterstützt:

Table 49. Untergeordnete Elemente

Untergeordnetes Element	Erforderlich	Beschreibung
<b>&lt;Command&gt;</b>	Nein	Befehl
<b>&lt;Argument&gt;</b>	Nein	Befehlszeilenargumente
<b>&lt;Stdin&gt;</b>	Nein	Vom Element <b>&lt;Stdin&gt;</b> unterstützte Argumente
<b>&lt;ErrorMessage&gt;</b>	Nein	Auszugebende Nachricht, wenn Voraussetzungen nicht erfüllt sind

**Anmerkung:** Das Element **<Command>** muss für jede Voraussetzung auf **<ParameterDef>**-Ebene, auf **<CfgMethod>**-Ebene oder in einem Element **<PrereqDef>** definiert werden.

### Verwendung

**Prereq**-Elemente sind Befehle, die Bedingungen für die Ausführung von **<Get>**-, **<Set>**- und **<Discover>**-Operationen für Parameter festlegen, die die **<Get>**-, **<Set>**- oder **<Discover>**-Operation verwenden. Parameter, für die ein **prereq**-Befehl scheitert (Rückgabecode ungleich null), werden ignoriert, und die in der Voraussetzung definierte Fehlernachricht wird angezeigt.

Das Element **<PrereqDef>** definiert eine Voraussetzung. Die Voraussetzungen können später mithilfe eines Elements **<Prereq>**, das dasselbe Attribut *id* hat, einer Operation eines Parameters oder einer Konfigurationsmethode zugewiesen werden.

### Beispiel

Im folgenden Beispiel wird die Voraussetzung *nodeId* definiert und den Parametern **netaddr** und **netaddr6** zugewiesen:

```
<PrereqDef id="nodeId">
 <Command>[[`usr/bin/uname -f` = %p[nodeId]]]</Command>
 <ErrorMessage>Parameter cannot be applied to a different node</ErrorMessage>
</PrereqDef>

<ParameterDef name="netaddr" type="string" targetClass="device" cfgmethod="attr">
 <Set type="permanent">
 <Prereq id="nodeId"/>
 </Set>
 <Property name="nodeId"/>
</ParameterDef>

<ParameterDef name="netaddr6" type="string" targetClass="device" cfgmethod="attr">
 <Set type="permanent">
 <Prereq id="nodeId"/>
 </Set>
 <Property name="nodeId"/>
</ParameterDef>
```

In diesem Beispiel wird der Test nur ein einziges Mal ausgeführt, weil die beiden Parameter dasselbe Element **<Command>** für ihre Voraussetzungen verwenden und die generierte Befehlszeile für die beiden Parameter identisch ist.

### Zugehörige Informationen

- „Befehlszeilengenerierung“ auf Seite 125
- „Element **<Prereq>**“ auf Seite 119

Element **<Property>**:

Das Element **<Property >** weist einem Parameter oder einer Konfigurationsmethode eine Eigenschaft zu.

### Syntax

Übergeordnetes Element: **<CfgMethod>**, **<ParameterDef>**

Das folgende Attribut wird unterstützt:

*Tabelle 50. Attribut*

Attribut	Erforderlich	Typ	Beschreibung
<i>name</i>	Ja	string	Gibt den Namen der Eigenschaft an.

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 51. Untergeordnetes Element

Untergeordnetes Element	Erforderlich	Beschreibung
<Command>	Nein	Befehl
<Argument>	Nein	Befehlszeilenargumente
<Stdin>	Nein	Vom Element <Stdin> unterstützte Argumente
<Filter>	Nein	Filter
<Mask>	Nein	Maske für Erfassung der Ausgabe

**Anmerkung:** Das Element <Command> muss für jede Eigenschaft auf <ParameterDef>-Ebene, auf <CfgMethod>-Ebene oder in einem Element <PropertyDef> definiert werden.

## Verwendung

Eigenschaften sind Schlüssel/Wert-Paare, die einem Parameter zugewiesen werden. Der Wert des Schlüssel/Wert-Paars wird mit den Befehlen `artexget -r` und `artexget -n` abgerufen und im Ausgabeprofil gespeichert. In einem Profil gespeicherte Eigenschaftswerte können mit der Sequenz `%p[Eigenschaftsname]` in eine Befehlszeile eingefügt werden.

Das Element <Property > weist einem Parameter oder einer Konfigurationsmethode eine Eigenschaft zu. Die Eigenschaft wird entweder lokal im Element <Property> definiert oder wird von einem Element <Property> oder <PropertyDef> höherer Ebene übernommen, das ein entsprechendes Attribut "name" hat.

Die Eigenschaften eines Parameters werden alle lokal im Element <ParameterDef> definiert. Außerdem werden alle Eigenschaften des Parameters in der Konfigurationsmethode des Parameters definiert, wenn eine Konfigurationsmethode verwendet wird. Wenn eine Eigenschaft in einem Element <CfgMethod> definiert wird, haben deshalb automatisch alle Elemente <ParameterDef>, die die Konfigurationsmethode verwenden, automatisch diese Eigenschaft (obwohl einige Elemente die Eigenschaft möglicherweise lokal neu definieren).

Eigenschaftswerte werden aus der Ausgabe einer Befehlszeile extrahiert. Die Befehlszeile wird erstellt, indem die Elemente <Command>, <Argument>, <Stdin> und <Filter> gemäß der Beschreibung im Abschnitt Befehlszeilengenerierung kombiniert werden. Sie müssen einen der folgenden Eigenschaftswerte verwenden: die unaufbereitete Ausgabe der Befehlszeile oder den Teil der Ausgabe, der der Maske entspricht, wenn ein Element <Mask> angegeben ist.

Die Elemente <Command>, <Argument>, <Stdin>, <Filter> und <Mask>, die eine Eigenschaft definieren, werden in der folgenden Reihenfolge gesucht:

- Im Element <Property> auf <ParameterDef>-Ebene
- Wenn das Element <ParameterDef> ein Attribut `cfgmethod` hat, in der Konfigurationsmethode des Elements <Property>, das ein entsprechendes Attribut `name` hat
- Im Element <PropertyDef> des Katalogs, der ein entsprechendes Attribut "name" hat

## Beispiel

Im folgenden Beispiel wird den Parametern `netaddr` und `netaddr6` die Eigenschaft `nodeId` zugewiesen:

```
<ParameterDef name="netaddr" type="string" targetClass="device" cfgmethod="attr">
 <Property name="nodeId">
 <Command>/usr/bin/uname -f/<Command>
 <Mask>.*/<Mask>
 </Property>
</ParameterDef>

<ParameterDef name="netaddr6" type="string" targetClass="device" cfgmethod="attr">
```

```

<Property name="nodeId">
 <Command>/usr/bin/uname -f</Command>
 <Mask>.*</Mask>
</Property>
</ParameterDef>

```

In diesem Beispiel entspricht die Maske der gesamten Zeile und wird nur verwendet, um das Zeilenvorschubzeichen am Ende der Befehlsausgabe auszuschließen.

In diesem Beispiel wird der Befehl **uname** zweimal ausgeführt: einmal für den Parameter **netaddr** und einmal für den Parameter **netaddr6**. Der Befehl wird zweimal ausgeführt, weil jeder Parameter eine eigene Eigenschaft mit einem eigenen Element **<Command>** hat. Im Abschnitt „Element **<PropertyDef>**“ finden Sie ein Beispiel, in dem der Befehl **uname** nur ein einziges Mal ausgeführt werden muss.

### Zugehörige Informationen

- „Befehlszeilengenerierung“ auf Seite 125
- „Erweiterung der Befehlszeilenelemente“ auf Seite 128
- „Element **<PropertyDef>**“

Element **<PropertyDef>**:

Das Element **<PropertyDef>** definiert eine Eigenschaft, die in einem Element **<Property>** verwendet werden kann.

### Syntax

Übergeordnetes Element: **<Catalog>**

Das folgende Attribut wird unterstützt:

Tabelle 52. Attribut

Attribut	Erforderlich	Typ	Beschreibung
<i>name</i>	Ja	string	Gibt den Namen der Eigenschaft an.

Die folgenden untergeordneten Elemente werden unterstützt:

Tabelle 53. Untergeordnetes Element

Untergeordnetes Element	Erforderlich	Beschreibung
<b>&lt;Command&gt;</b>	Nein	Befehl
<b>&lt;Argument&gt;</b>	Nein	Befehlszeilenargumente
<b>&lt;Stdin&gt;</b>	Nein	Vom Element <b>&lt;Stdin&gt;</b> unterstützte Argumente
<b>&lt;Filter&gt;</b>	Nein	Filter
<b>&lt;Mask&gt;</b>	Nein	Maske für Erfassung der Ausgabe

**Anmerkung:** Das Element **<Command>** muss für jede Eigenschaft auf **<ParameterDef>**-Ebene, auf **<Cfg-Method>**-Ebene oder in einem Element **<PropertyDef>** definiert werden.

### Verwendung

Eigenschaften sind Schlüssel/Wert-Paare, die einem Parameter zugewiesen werden. Der Wert des Schlüssel/Wert-Paars wird mit den Befehlen **artexget -r** und **artexget -n** abgerufen und im Ausgabeprofil gespeichert. In einem Profil gespeicherte Eigenschaftswerte können mit der Sequenz **%p[Eigenschaftsname]** in eine Befehlszeile eingefügt werden.

Das Element **<PropertyDef>** definiert eine Eigenschaft. Diese Eigenschaft kann später mithilfe eines Elements **<Property>**, das dasselbe Attribut "name" hat, einem Parameter oder einer Konfigurationsmethode zugewiesen werden.

### Beispiel

Im folgenden Beispiel wird den Parametern **netaddr** und **netaddr6** die Eigenschaft *nodeId* zugewiesen:

```
<PropertyDef name="nodeId">
 <Command>/usr/bin/uname -f</Command>
 <Mask>.*</Mask>
</PropertyDef>

<ParameterDef name="netaddr" type="string" targetClass="device" cfgmethod="attr">
 <Property name="nodeId"/>
</ParameterDef>

<ParameterDef name="netaddr6" type="string" targetClass="device" cfgmethod="attr">
 <Property name="nodeId"/>
</ParameterDef>
```

In diesem Beispiel wird der Befehl **uname** nur ein einziges Mal ausgeführt, weil die beiden Parameter dasselbe Element **<Command>** für ihre Eigenschaften verwenden und die generierte Befehlszeile für die beiden Parameter identisch ist.

### Zugehörige Informationen

- „Befehlszeilengenerierung“
- „Erweiterung der Befehlszeilenelemente“ auf Seite 128
- „Element **<Property>**“ auf Seite 122

### Befehlszeilengenerierung

Das Framework AIX Runtime Expert stützt sich auf externe Befehle, die Parameterwerte erfassen, setzen und (optional) vergleichen. In diesem Abschnitt wird beschrieben, wie Befehlszeilen basierend auf den in den Katalogdateien bereitgestellten Syntaxinformationen erstellt werden.

### Operationen

Für jeden Parameter können die folgenden Operationen definiert werden:

- Get **type="current"**: Wird verwendet, um den aktuellen Wert des Parameters zu erfassen.
- Get **type="nextboot"**: Wird verwendet, um den Wert des Parameters zu erfassen, den der Parameter nach einem Warmstart hat.
- Set **type="current"**: Wird verwendet, um den aktuellen Wert des Parameters zu setzen. Dieser Parameterwert geht bei einem Warmstart verloren.
- Set **type="nextboot"**: Wird verwendet, um den Wert des Parameters zu setzen, den der Parameter nach einem Warmstart hat.
- Set **type="permanent"**: Wird verwendet, um den aktuellen Wert des Parameters zu setzen, den der Parameter auch nach einem Warmstart beibehält.
- **diff**: Wird verwendet, um zwei Werte des Parameters miteinander zu vergleichen.
- Discover: Wird verwendet, um Ziele für Parameter zu suchen, die diese Parameter unterstützen.
- Eigenschaft, die zum Erfassen einer Eigenschaft für einen Parameter verwendet wird
- Voraussetzung, die verwendet wird, um Bedingungen für die Ausführung einer get-, set- oder discover-Operation für einen bestimmten Parameter festzulegen

Es müssen nicht alle Operationen für alle Parameter definiert werden. Die beiden **get**-Operationen und alle **set**-Operationen, die von den Parametern unterstützt werden, müssen definiert werden. Die Operation **diff** ist optional, und wenn sie nicht definiert wird, werden Vergleiche von Parameterwerten intern

auf der Basis des Parametertyps, wie z. B. string oder integer, vorgenommen. Die Operation **discover** darf nur für Parameter definiert werden, die ein Ziel haben. Eigenschaften und Voraussetzungen werden nur bei Bedarf definiert.

## Befehlszeilenelemente

Für jede von einem Parameter unterstützte Operation können bis zu fünf verschiedene Elemente verwendet werden, um zu definieren, wie eine Befehlszeile zum Ausführen der Operation erstellt werden kann:

- **<Command>**: Wird verwendet, um den Basisbefehl für die Behandlung der Parameter zu definieren.
- **<Stdin>**: Wird verwendet, um die Daten zu definieren, die in die Standardeingabe der Befehlszeile geschrieben wird.
- **<Argument>**: Wird verwendet, um parameterspezifische Daten in ein Element **<Command>** oder **<Stdin>** einzufügen.
- **<Filter>**: Wird verwendet, um die Ausgabe einer Befehlszeile für die **get**- und **diff**-Operationen zu filtern.
- **<Mask>**: Wird verwendet, um Daten aus der Ausgabe einer Befehlszeile für die **get**-, **diff**- und **property**-Operationen zu extrahieren.

Wenn eine Operation ausgeführt werden muss, werden die für die angeforderte Operation definierten Elemente **<Command>**, **<Stdin>**, **<Argument>** und **<Filter>** zu einer Gruppe von Befehlszeilen kombiniert, wie im Abschnitt „Algorithmus für die Befehlszeilengenerierung“ auf Seite 127 erläutert wird. Die generierten Befehlszeilen werden von einer Shell ausgeführt. Für die **get**-, **diff**- und **property**-Operationen wird das Element **<Mask>** verwendet, um die angeforderten Daten (Parameterwerte, Vergleichsergebnisse oder Eigenschaftswerte) aus der Befehlszeile zu extrahieren.

## Konfigurationsmethoden

Befehlszeilenelemente können lokal in einem Element **<ParameterDef>** definiert oder von einem Element **<CfgMethod>** übernommen werden, das im Element **<ParameterDef>** mit dem Attribut *cfgmethod* referenziert wird.

Die Kombination der folgenden Komponenten ist zulässig: Die Gruppe der Befehlszeilenelemente, die für eine bestimmte Operation eines bestimmten Parameters definiert sind, ist die Union der lokal im Element **<ParameterDef>** definierten Befehlszeilenelemente und der Befehlszeilenelemente, die für dieselbe Operation im Element **<CfgMethod>** definiert sind, das vom Attribut *cfgmethod* des Elements **<ParameterDef>** referenziert wird. Wenn dasselbe Befehlszeilenelement lokal und in einer Konfigurationsmethode definiert ist, hat die lokale Definition Vorrang.

Im Folgenden sehen Sie eine nicht optimierte Katalogdatei:

```
<CfgMethod id="vmo">
 <Get type="nextboot">
 <Command>/usr/sbin/vmo -r%a</Command>
 <Mask name="1" value="2">[[[:space:]]*(.*) = (.*)</Mask>
 </Get>

 <Set type="permanent">
 <Command>/usr/sbin/vmo -p -o%a</Command>
 <Argument> -o %n=%p</Argument>
 </Set>
</CfgMethod>

<ParameterDef name="lgpg_size" cfgmethod="vmo">
 <Get type="current">
 <Command>/usr/sbin/vmo -o lgpg_size</Command>
 <Mask name="1" value="2">[[[:space:]]*(.*) = (.*)</Mask>
 </Get>
```

```

<Get type="nextboot">
 <Argument> -o lgpg_size</Argument>
</Get>

</ParameterDef>

```

Erläuterungen zu dieser Datei:

- Die Operation `<Get type="current">` wird vollständig auf der Ebene von `<ParameterDef>` definiert.
- Die Operation `<Get type="nextboot">` hat einige Elemente, die auf der Ebene von `<CfgMethod>` definiert sind (`<Command>` und `<Mask>`), und einige Elemente, die auf der Ebene von `<ParameterDef>` definiert sind (`<Argument>`).
- Die Operation `<Get type="current">` wird vollständig auf der Ebene von `<CfgMethod>` definiert.

Die Verwendung einer Konfigurationsmethode hat die folgenden beiden Vorteile:

- Sie vereinfacht den Katalog. In vielen Fällen übernehmen Parameterdefinitionen alle Befehlszeilenelemente von einer Konfigurationsmethode, und das Element `<ParameterDef>` ist leer.
- Sie lässt die Gruppierung verschiedener Parameter in einer einzigen Befehlszeile zu, sofern dies möglich ist.

## Algorithmus für die Befehlszeilengenerierung

Befehlszeilen werden mit einem Algorithmus generiert, der die Gruppierung mehrerer Parameter in einem einzigen Befehl zulässt.

Die Parametergruppierung ist nicht nur aus Leistungs- und Effizienzgründen, sondern auch für bestimmte Parameter erforderlich. Die `vmo`-Parameter `lgpg_regions` und `lgpg_size` können beispielsweise nicht unabhängig voneinander gesetzt werden, sondern müssen gemeinsam in einem einzigen Befehlsaufruf `vmo` gesetzt werden.

Der Algorithmus für die Befehlszeilengenerierung entspricht funktional den folgenden Schritten:

1. Die Elemente `<Command>` und `<Stdin>` jedes Parameters im Eingabeprofil werden teilweise erweitert. In dieser Phase werden die Sequenzen `%a`, `%v1[name]`, `%v2[name]`, `%f1[name]` und `%f2[name]` ignoriert und nicht erweitert.
2. Parameter, die alle fünf folgenden Bedingungen prüfen, werden gruppiert:
  - Parameter verwenden dasselbe Element `<Command>`.
  - Parameter verwenden dasselbe Element `<Stdin>`.
  - Parameter verwenden dasselbe Element `<Filter>`.
  - Die Erweiterung des Elements `<Command>`, die in Schritt 1 durchgeführt wurde, hat identische Zeichenfolgen erzeugt.
  - Die Erweiterung des Elements `<Stdin>`, die in Schritt 1 durchgeführt wurde, hat identische Zeichenfolgen erzeugt.

Die Gruppe hat jetzt eigene, teilweise erweiterte Elemente `<Command>` und `<Stdin>` sowie ein eigenes Element `<Filter>`, das von allen Parametern in der Gruppe gemeinsam genutzt wird.
3. Für jede Parametergruppe werden die Sequenzen `%v1[name]`, `%v2[name]`, `%f1[name]` und `%f2[name]` der Elemente `<Command>` und `<Stdin>` der Gruppe erweitert. Der Parametername wird nur in der Gruppe gesucht.
4. Für jede Parametergruppe wird die Sequenz `%a` für die Elemente `<Command>` und `<Stdin>` der Gruppe erweitert: Für jeden Parameter in der Gruppe wird das Element `<Argument>` erweitert, und die Verkettung dieser erweiterten Elemente `<Argument>` ersetzt jede Sequenz `%a` in den Elementen `<Command>` und `<Stdin>`.

Das Ergebnis dieses Prozesses ist eine Gruppe von Befehlszeilen mit optionalen Daten, die in die Standardeingabe geschrieben werden sollen, und einem Befehl zum Filtern der Ausgabe.

## Erweiterung der Befehlszeilenelemente:

Die Elemente **<Command>**, **<Stdin>** und **<Argument>** unterstützen spezielle Sequenzen, die vom Framework AIX Runtime Expert zu den endgültigen Befehlszeilen erweitert werden.

Die folgende Tabelle ist eine Kurzreferenz aller unterstützten Sequenzen. Weitere Einzelheiten zu einer Sequenz finden Sie in den folgenden Abschnitten.

Tabelle 54. Sequenz

Sequenz	Erweiterung zu
%%	Literalzeichen %
%a	Verkettung der erweiterten <b>Argumentzeichenfolgen</b> für alle Parameter, die in derselben Befehlszeile verarbeitet werden können.
%n	Name des Parameters
%v1	Wert des Parameters
%v2	Zweiter Wert des Parameters (nur gültig für <b>diff</b> -Operationen)
%f1	Name der temporären Datei, die den Wert des Parameters enthält
%f2	Name der temporären Datei, die den zweiten Wert des Parameters enthält (nur gültig für <b>diff</b> -Operationen)
%v1[name]	Wert des Parameters "name"
%v2[name]	Zweiter Wert des Parameters "name" (nur gültig für <b>diff</b> -Operationen)
%f1[name]	Name einer temporären Datei, die den Wert des Parameters "name" enthält
%f2[name]	Name der temporären Datei, die den zweiten Wert des Parameters "name" enthält (nur gültig für <b>diff</b> -Operationen)
%t[class]	Name der Zielinstanz für die Zielklasse
%p[name]	Wert der Eigenschaft <i>Name</i>
%c	Katalog-ID

## Verwendung von Escapezeichen für %-Sequenzen

Parameternamen, Parameterwerte und Zielnamen, die von von AIX Runtime Expert erweitert werden, werden in einfache Anführungszeichen eingeschlossen, wenn sie in einem Element **<Command>** oder in einem Element **<Argument>** verwendet werden, das (über die Sequenz %a) in ein Element **<Command>** eingefügt werden soll. Auf diese Weise soll sichergestellt werden, dass diese Zeichenfolgen als Einzelwort an die Shell übergeben werden, selbst wenn sie Leerzeichen oder andere Sonderzeichen enthalten. Außerdem wird allen einfachen Anführungszeichen im erweiterten Ausdruck ein entsprechendes Escapezeichen vorangestellt.

Die Katalogautoren dürfen die Sequenzen %n, %v1, %v2, %v1[name], %v2[name] und %t[class] nicht in einer Zeichenfolge in Anführungszeichen verwenden. Wenn diese Sequenzen in einer Zeichenfolge verwendet werden müssen, muss die Zeichenfolge vor der %-Sequenz geschlossen werden, wie im folgenden Beispiel gezeigt:

```
echo "Parameter "%n" is set to "%v1
```

Wenn die Zeichenfolge nicht vor der %-Sequenz geschlossen wird, werden ungültige Befehlszeilen erstellt, die ein Sicherheitsrisiko darstellen.



## Sequenz %%

Die Sequenz %% wird zum Literalzeichen % erweitert.

Beispiel:

```
/bin/ps -aeF"%%a"
```

Diese Zeichenfolge wird zu der folgenden Zeichenfolge erweitert:

```
/bin/ps -aeF"%a"
```

## Sequenz %a

Die Sequenz %a kann in der <Command>- oder in der <Stdin>-Zeichenfolge verwendet werden. Sie wird durch die Verkettung aller erweiterten <Argument>-Zeichenfolgen aller Parameter ersetzt, die in demselben Befehl erstellt werden können (eine formale Beschreibung der Parametergruppierungen finden Sie im Abschnitt Befehlszeilengenerierung).

Beispielkatalog (beachten Sie, dass dieser Katalog mit der Sequenz %n vereinfacht werden kann):

```
<CfgMethod id="vmo">
 <Get type="current">
 <Command>/usr/sbin/vmo%a</Command>
 </Get>
</CfgMethod>
<ParameterDef name="lgpg_size" cfgmethod="vmo">
 <Get type="current">
 <Argument> -o lgpg_size</Argument>
 </Get>
</ParameterDef>
<ParameterDef name="lgpg_regions" cfgmethod="vmo">
 <Get type="current">
 <Argument> -o lgpg_regions</Argument>
 </Get>
</ParameterDef>
```

Beispielprofil:

```
<Parameter name="lgpg_size" />
<Parameter name="lgpg_regions" />
```

Dieser Katalog und dieses Profil erzeugen die folgende Befehlszeile für die Operation "get current":

```
/usr/sbin/vmo -o lgpg_size -o lgpg_regions
```

## Sequenz %n

Die Sequenz %n wird durch den Namen des Parameters ersetzt.

Mit der Sequenz %n kann das Beispiel aus dem Abschnitt %a wie folgt vereinfacht werden:

```
<CfgMethod id="vmo">
 <Get type="current">
 <Command>/usr/sbin/vmo%a</Command>
 <Argument> -o %n</Argument>
 </Get>
</CfgMethod>
<ParameterDef name="lgpg_size" cfgmethod="vmo" />
<ParameterDef name="lgpg_regions" cfgmethod="vmo" />
```

Mit dem folgenden Profil:

```
<Parameter name="lgpg_size" />
<Parameter name="lgpg_regions" />
```

Die folgende Befehlszeile wird für die Operation "get current" generiert:

```
/usr/sbin/vmo -o 'lpgg_size' -o 'lpgg_regions'
```

### Sequenzen %v1 und %v2

Die Sequenz %v1 wird mit dem Wert des Parameters ersetzt.

Die Sequenz %v2 ist nur für <Diff>-Operationen gültig und wird mit dem zweiten Wert des Parameters ersetzt.

Beispielkatalog:

```
<CfgMethod id="vmo">
 <Set type="permanent">
 <Command>/usr/sbin/vmo -p%a</Command>
 <Argument> -o %n=%v1</Argument>
 </Set>
</CfgMethod>
<ParameterDef name="lpgg_size" cfgmethod="vmo" />
<ParameterDef name="lpgg_regions" cfgmethod="vmo" />
```

Beispielprofil:

```
<Parameter name="lpgg_size" value="16M"/>
<Parameter name="lpgg_regions" value="128" />
```

Mit diesem Katalog und diesem Profil wird die folgende Befehlszeile für die Operation **set permanent** generiert:

```
/usr/sbin/vmo -p -o 'lpgg_size'='16M' -o 'lpgg_regions'='128'
```

### Sequenzen %f1 und %f2

Die Sequenzen %f1 und %f2 werden durch den Namen der temporären Datei ersetzt, die vor der Ausführung des Befehls erstellt wurde. Der Dateinhalt ist der Wert des Parameters für %f1 und der zweite Wert des Parameters für %f2. Die Sequenz %f2 kann nur für <Diff>-Operationen verwendet werden.

Beispielkatalog:

```
<ParameterDef name="some_file">
 <Diff>
 <Command>/usr/bin/diff %f1 %f2</Command>
 </Diff>
</ParameterDef>
```

Der Befehl **artexdiff** wird zwischen den beiden Profilen ausgeführt, die denselben Parameter mit einem jeweils anderen Wert enthalten:

```
<Parameter name="some_file" value="foo" />
<Parameter name="some_file" value="bar" />
```

In diesem Fall werden zwei temporäre Dateien, /tmp/file1 und /tmp/file2 (die tatsächlichen Dateinamen sind anders), mit den Zeichenfolgen "foo" und "bar" erstellt, und der folgende Befehl wird ausgeführt:

```
/usr/bin/diff /tmp/file1 /tmp/file2
```

### Sequenzen %v1[name] und %v2[name]

Die Sequenz %v1[name] wird durch den Wert des Parameternamens ersetzt.

Die Sequenz %v2[name] ist nur für <Diff>-Operationen gültig und wird durch den zweiten Wert des Parameternamens ersetzt.

Diese Sequenzen sind hilfreich, wenn ein Konfigurationsbefehl mehrere Parameter gleichzeitig akzeptiert, aber erfordert, dass einige dieser Parameter an eine bestimmte Position in der Befehlszeile gesetzt werden. Dies gilt beispielsweise für den Befehl **chcons**, der erfordert, dass der Pfad zur Konsoleneinheit oder die Datei als letztes Element in der Befehlszeile stehen. Mit der Sequenz `%v1[name]` kann der Katalog **chcons** wie folgt geschrieben werden:

```
<CfgMethod id="chcons">
 <Set type="nextboot">
 <Command>/usr/sbin/chcons%a %v1[console_device]</Command>
 <Argument> -a %n=%v1</Argument>
 </Set>
</CfgMethod>
<ParameterDef name="console_device" cfgmethod="chcons" reboot="true" />
<ParameterDef name="console_logname" cfgmethod="chcons" reboot="true" />
<ParameterDef name="console_logsize" cfgmethod="chcons" reboot="true" />
```

Beispielprofil:

```
<Parameter name="console_device" value="/dev/vty0"/>
<Parameter name="console_logname" value="/var/adm/ras/conslog" />
<Parameter name="console_logverb" value="9" />
```

Dieser Katalog generiert die folgende Befehlszeile für die Operation **set nextboot**:

```
/usr/sbin/chcons -a 'console_logname'='/var/adm/ras/conslog' -a 'console_logverb'='9' /dev/vty0
```

### Sequenzen `%f1[name]` und `%f2[name]`

Die Sequenzen `%f1[name]` und `%f2[name]` werden durch den Namen der temporären Datei ersetzt, die vor der Ausführung des Befehls erstellt wurde. Der Dateinhalt ist der Wert des Parameternamens für `%f1[name]` und der zweite Wert des Parameternamens für `%f2[name]`. Die Sequenz `%f2[name]` kann nur für **<Diff>**-Operationen verwendet werden.

### Sequenzen `%t[class]`

Die Sequenz `%t[class]` wird durch den Namen der Zielinstanz ersetzt, die als Zielklasse behandelt wird.

Die Sequenz `%t[class]` wird für Parameter verwendet, die für ein bestimmtes Objekt und nicht für das gesamte System gelten. Ein Beispiel ist der Befehl **chuser**, dessen Parameter für einen bestimmten Benutzer (`root`, `guest`) für eine bestimmte Registry (`files`, `LDAP`) gelten. Der Katalog für den Befehl **chuser** muss wie folgt geschrieben werden:

```
<CfgMethod id="chuser">
 <Set type="permanent">
<Command>/usr/bin/chuser -R %t[module]%a %t[user]</Command>
 <Argument> %n=%v1</Argument>
 </Set>
</CfgMethod>
<ParameterDef name="shell" cfgmethod="chuser" targetClass="module,user">
<ParameterDef name="histsize" cfgmethod="chuser" targetClass="module,user" />
```

Sehen Sie sich das folgende Profil an, das die `Shell`- und `histsize`-Parameter für die Benutzer *adam* und *bob* in den `LDAP`- und `files`-Registries setzt:

```
<Parameter name="shell" value="/usr/bin/ksh">
 <Target class="module" instance="LDAP" />
 <Target class="user" instance="adam" />
</Parameter>
<Parameter name="histsize" value="5000">
 <Target class="module" instance="LDAP" />
 <Target class="user" instance="adam" />
</Parameter>
<Parameter name="shell" value="/usr/bin/ksh">
 <Target class="module" instance="files" />
 <Target class="user" instance="adam" />
```

```

</Parameter>
<Parameter name="histsize" value="5000">
 <Target class="module" instance="files" />
 <Target class="user" instance="adam" />
</Parameter>
<Parameter name="shell" value="/usr/bin/bash">
 <Target class="module" instance="LDAP" />
 <Target class="user" instance="bob" />
</Parameter>
<Parameter name="histsize" value="10000">
 <Target class="module" instance="LDAP" />
 <Target class="user" instance="bob" />
</Parameter>
<Parameter name="shell" value="/usr/bin/bash">
 <Target class="module" instance="files" />
 <Target class="user" instance="bob" />
</Parameter>
<Parameter name="histsize" value="10000">
 <Target class="module" instance="files" />
 <Target class="user" instance="bob" />
</Parameter>

```

In diesem Fall werden die folgenden Befehle ausgeführt:

```

/usr/bin/chuser -R 'LDAP' 'shell'='/usr/bin/ksh' 'histsize'='5000' 'adam'
/usr/bin/chuser -R 'files' 'shell'='/usr/bin/ksh' 'histsize'='5000' 'adam'
/usr/bin/chuser -R 'LDAP' 'shell'='/usr/bin/bash' 'histsize'='10000' 'bob'
/usr/bin/chuser -R 'files' 'shell'='/usr/bin/bash' 'histsize'='10000' 'bob'

```

Es werden vier Befehle generiert. Der Grund dafür ist, dass die Sequenzen `%t[module]` und `%t[user]` in der Zeichenfolge `<Command>` verwendet wurden, d. h., dass jeder Befehl für ein bestimmtes Modul und einen bestimmten Benutzer gilt. Deshalb werden nur Parameter, die für dasselbe Modul und denselben Benutzer gelten, gruppiert.

### Sequenz `%p[name]`

Die Sequenz `%p[name]` wird durch den Wert ersetzt, der im Eingabeprofil als Eigenschaftsname angegeben ist. Die folgende Voraussetzung verwendet beispielsweise die Sequenz `%p[nodeId]`, um zu prüfen, ob die Knoten-ID des lokalen Systems (die vom Befehl `uname -f` zurückgegeben wird) mit der Knoten-ID übereinstimmt, die in der Eigenschaft `nodeId` des Profils gespeichert ist:

```

<PrereqDef id="nodeId">
 <Command>[[` /usr/bin/uname -f` = %p[nodeId]]]</Command>
 <ErrorMessage>Parameter cannot be applied to a different node</ErrorMessage>
</PrereqDef>

```

### Sequenz `%c`

Die Sequenz `%c` wird durch die ID der Katalogdatei ersetzt, zu der der Parameter gehört. Dies ist die im Profil angegebene Katalog-ID, die von der ID des Katalogs abweichen kann, der den Parameter tatsächlich definiert, wenn die Katalogübernahme verwendet wird.

Die folgende Voraussetzung verwendet beispielsweise die Sequenz `%c`, um zu prüfen, ob der `uniquetype`-Wert der Zieleinheit mit dem Namen der Katalogdatei übereinstimmt:

```

<PrereqDef id="devUniqueType">
 <Command>[["devParam.~/usr/sbin/ldev -F uniquetype -l %t[device] | /usr/bin/tr / .`" = %c]]</Command>
 <ErrorMessage>Parameter cannot be applied to a different device type</ErrorMessage>
</PrereqDef>

```

## Befehle und Prozesse

Ein *Befehl* ist eine Anforderung zur Ausführung einer Operation oder eines Programms. Ein *Prozess* ist ein Programm oder ein Befehl, der tatsächlich auf dem Computer ausgeführt wird.

Mit Befehlen wird das Betriebssystem angewiesen, bestimmte Tasks auszuführen. Befehle werden nach der Eingabe vom Befehlsinterpreter (auch *Shell* genannt) entschlüsselt. Anschließend wird die entsprechende Task ausgeführt.

Das Betriebssystem kann viele verschiedene Prozesse gleichzeitig verarbeiten.

Im Betriebssystem AIX können Sie die Eingabe und Ausgabe (E/A) von Daten mit bestimmten E/A-Befehlen und -Symbolen steuern. Sie können die Eingabe steuern, indem Sie angeben, wo die Daten erfasst werden sollen. Beispielsweise können Sie festlegen, dass auf der Tastatur eingegebene Daten (Standardeingabe) oder Daten aus einer Datei gelesen werden sollen. Sie können die Ausgabe steuern, indem Sie angeben, wo die Daten angezeigt oder gespeichert werden sollen. Beispielsweise können Sie festlegen, dass die Daten auf dem Bildschirm ausgegeben (Standardausgabe) oder in eine Datei geschrieben werden sollen.

## Befehle

Manche Befehle können durch Eingabe eines einzigen Wortes ausgeführt werden. Es ist auch möglich, Befehle zu kombinieren, um die Ausgabe eines Befehls als Eingabe für einen anderen Befehl zu verwenden.

Das Kombinieren von Befehlen, so dass die Ausgabe eines Befehls als Eingabe für einen anderen Befehl verwendet wird, wird als *Verkettung* (Piping) bezeichnet.

Mit Flags werden die von Befehlen auszuführenden Aktionen näher definiert. Ein *Flag* ist ein Änderungswert, der zusammen mit dem Befehlsnamen in der Befehlszeile eingegeben wird. Vor dem Flag wird gewöhnlich ein Minuszeichen eingegeben.

Befehle können in Gruppen zusammengefasst und in einer Datei gespeichert werden. Diese Dateien werden als *Shellprozeduren* oder *Shell-Scripts* bezeichnet. Anstelle der einzelnen Befehle wird die Datei, die die Befehle enthält, ausgeführt.

Zum Ausführen eines Befehls geben Sie an der Eingabeaufforderung den Befehlsnamen ein. Drücken Sie dann die Eingabetaste.

```
$ Befehlsname
```

### Zugehörige Konzepte:

„Shellfunktionen“ auf Seite 221

Die Verwendung der Shell als Schnittstelle zum System hat verschiedene Vorteile.

### Zugehörige Tasks:

„Shell-Script erstellen und ausführen“ auf Seite 224

Ein *Shell-Script* ist eine Datei, die einen oder mehrere Befehl(e) enthält. Shell-Scripts sind eine einfache Möglichkeit, komplexe Befehle, lange oder komplizierte Befehlsfolgen und Routineaufgaben auszuführen. Wenn der Benutzer den Namen einer Shell-Scriptdatei eingibt, führt das System die in der Datei enthaltene Befehlsfolge aus.

### Befehlssyntax und Befehlsnamen:

Für manche Befehle muss nur ein Wort eingegeben werden, andere Befehle hingegen erfordern Flags und Parameter. Jeder Befehl hat eine Syntax, die die erforderlichen und optionalen Flags und Parameter festlegt.

Das allgemeine Befehlsformat ist wie folgt:

```
Befehlsname Flag(s) Parameter
```

Im Folgenden sind einige allgemeine Regeln für Befehle aufgelistet:

- Leerzeichen zwischen Befehlen, Flags und Parametern sind wichtig.

- Es ist möglich, zwei Befehle in derselben Zeile einzugeben. Die Befehle müssen durch ein Semikolon (;) voneinander getrennt werden. Beispiel:

```
$ Befehl1;Befehl2
```

Die Befehle werden nacheinander von der Shell verarbeitet.

- Bei der Eingabe von Befehlen muss die Groß-/Kleinschreibung beachtet werden. Die Shell unterscheidet zwischen Groß- und Kleinbuchstaben. Für die Shell ist der Befehl print nicht identisch mit dem Befehl PRINT oder Print.
- Lange Befehle können über mehrere Zeilen verteilt werden. Verwenden Sie dazu am Ende jeder Zeile einen Backslash (\). Der Backslash zeigt der Shell die Zeilenfortsetzung an. Der folgende Beispielbefehl erstreckt sich über zwei Zeilen:

```
$ ls Mail info temp \
(Eingabetaste drücken)
```

```
> diary
(die Eingabeaufforderung > erscheint)
```

Das Zeichen > ist die Sekundäreingabeaufforderung (\$ ist die voreingestellte Primäreingabeaufforderung) und zeigt an, dass die aktuelle Zeile die Fortsetzung der vorherigen ist. Beachten Sie, dass csh (C-Shell) keine Sekundäreingabeaufforderung ausgibt und der Umbruch an einer Wortgrenze erfolgen muss. Die Primäreingabeaufforderung der C-Shell ist %.

Das erste Wort jedes Befehls ist der Befehlsname. Einige Befehle haben nur einen Befehlsnamen.

*Befehls-Flags:*

Dem Befehlsnamen kann eine Reihe von Flags folgen. Flags modifizieren die Ausführung eines Befehls und werden mitunter auch als *Optionen* bezeichnet.

Flags beginnen immer mit einem Minuszeichen (-) und werden durch Leerzeichen oder Tabulatorzeichen voneinander getrennt. Ausnahmen sind die Befehle **ps**, **tar** und **ar**. Einigen Flags dieser Befehle muss kein Minuszeichen vorangestellt werden. Beispiel:

```
ls -a -F
```

ls ist der Befehlsname, und **-a -F** sind die Flags.

Werden in einem Befehl Flags verwendet, müssen diese direkt hinter dem Befehlsnamen angegeben werden. In einem Befehl können Flags, die aus einem Zeichen bestehen, mit einem Minuszeichen kombiniert werden. Der Befehl aus dem vorherigen Beispiel kann auch folgendermaßen eingegeben werden:

```
ls -aF
```

Manchmal beginnt ein Parameter tatsächlich mit einem Minuszeichen (-). In diesen Fällen müssen zwei Minuszeichen (--) als Begrenzer vor dem Parameter angegeben werden. Der Begrenzer -- zeigt dem Befehl an, dass es sich bei den weiteren Angaben nicht um ein Flag, sondern um einen Parameter handelt.

Wenn Sie beispielsweise ein Verzeichnis mit dem Namen -tmp erstellen möchten und den Befehl

```
mkdir -tmp
```

eingeben, wird eine Fehlermeldung wie die folgende angezeigt:

```
mkdir: Nicht erkanntes Flag: t
Verwendung: mkdir [-p] [-m Modus] Verzeichnis ...
```

Die korrekte Syntax für den Befehl ist:

```
mkdir -- -tmp
```

Dieser Befehl erstellt das neue Verzeichnis -tmp.

### *Befehlsparameter:*

Dem Befehlsnamen und den Flags können verschiedene Parameter folgen. Parameter werden manchmal auch als *Argumente* oder *Operanden* bezeichnet. Mit Parametern werden Informationen angegeben, die für die Ausführung des Befehls erforderlich sind.

Wenn Sie keinen Parameter angeben, verwendet der Befehl möglicherweise einen Standardwert. Beispiel:

```
ls -a temp
```

`ls` ist der Befehlsname, `-a` das Flag und `temp` der Parameter. Dieser Befehl zeigt alle (`-a`) Dateien im Verzeichnis `temp` an.

Im folgenden Beispiel ist das aktuelle Verzeichnis der Standardwert, weil keine Parameter angegeben sind:

```
ls -a
```

Der Standardwert ist das aktuelle Verzeichnis, weil kein Parameter angegeben ist.

Im folgenden Beispiel sind keine Flags angegeben, und `temp` und `mail` sind Parameter:

```
ls temp mail
```

In diesem Fall sind `temp` und `mail` zwei unterschiedliche Verzeichnisnamen. Der Befehl `ls` zeigt alle Dateien (mit Ausnahme der verdeckten) in diesen beiden Verzeichnissen an.

Wenn ein Parameter oder Optionsargument ein numerischer Wert ist oder einen solchen enthält, wird der Wert als dezimale Ganzzahl interpretiert, sofern keine andere Angabe erfolgt. Numerische Werte zwischen 0 und `INT_MAX`, die in der Datei `/usr/include/sys/limits.h` definiert sind, werden syntaktisch als numerische Werte erkannt.

Wenn ein Befehl, den Sie verwenden möchten, negative Zahlen als Parameter oder Optionsargument akzeptiert, können Sie Numerale aus dem Bereich `INT_MIN` bis `INT_MAX` verwenden, die in der Datei `/usr/include/sys/limits.h` definiert sind. Dies bedeutet aber nicht zwingend, dass alle Zahlen aus diesem Bereich semantisch korrekt sind. Einige Befehle besitzen integrierte Spezifikationen, die nur einen kleineren Zahlenbereich zulassen (z. B. einige Druckbefehle). Wenn ein Fehler generiert wird, informiert Sie die Fehlermeldung darüber, dass der Wert außerhalb des unterstützten Bereichs liegt, und nicht, dass der Befehl syntaktisch nicht korrekt ist.

### **Verwendungsanweisungen:**

Verwendungsanweisungen sind eine Methode für die Darstellung der Befehlsyntax und enthalten Symbole wie eckige Klammern, (`[ ]`), geschweifte Klammern (`{ }`) und vertikale Balken (`|`).

Das folgende Beispiel veranschaulicht die Verwendung des Befehls `unget`:

```
unget [-rSID] [-s] [-n] Datei ...
```

Die folgenden Konventionen gelten für die Verwendungsanweisungen für Befehle:

- Elemente, die unverändert in der Befehlszeile angegeben werden müssen, werden in **Fettschrift** angezeigt. Zu diesen Elementen gehören der Befehlsname, Flags und Literalzeichen.
- Elemente, die für Variablen stehen und durch einen Namen ersetzt werden müssen, werden in *Kursivschrift* angezeigt. Zu diesen Elementen gehören Parameter, die Flags folgen, und Parameter, die der Befehl liest, z. B. *Dateien* und *Verzeichnisse*.
- Parameter in eckigen Klammern sind optional.
- Parameter in geschweiften Klammern sind erforderlich.

- Nicht in Klammern eingeschlossene Parameter sind erforderlich.
- Ein vertikaler Balken bedeutet, dass nur ein Parameter ausgewählt werden kann. [ a | b ] gibt beispielsweise an, dass Sie a oder b oder keinen dieser beiden Parameter auswählen *können*. { a | b } gibt an, dass einer der beiden Parameter (a oder b) ausgewählt werden *muss*.
- Auslassungszeichen ( ... ) zeigen an, dass der Parameter in der Befehlszeile wiederholt werden kann.
- Das Minuszeichen ( - ) steht für die Standardeingabeinheit.

### **Befehl Shutdown:**

Wenn Sie Rootberechtigung besitzen, können Sie das System mit dem Befehl **shutdown** herunterfahren. Sollten Sie nicht zum Ausführen des Befehls **shutdown** berechtigt sein, melden Sie sich einfach vom Betriebssystem ab und lassen Sie das System eingeschaltet.

**Achtung:** Schalten Sie das System nicht aus, ohne zuvor einen Systemabschluss durchgeführt zu haben. Durch das Ausschalten des Systems werden alle auf dem System gestarteten Prozesse beendet. Sollten andere Benutzer auf dem System arbeiten oder Jobs im Hintergrund ausgeführt werden, könnten Daten verloren gehen. Sie müssen das System ordnungsgemäß herunterfahren, bevor Sie es stoppen.

Geben Sie an der Eingabeaufforderung Folgendes ein:

```
shutdown
```

Wenn die Ausführung des Befehls **shutdown** abgeschlossen und das Betriebssystem heruntergefahren ist, wird folgende Nachricht angezeigt:

```
....Shutdown completed....
```

Die vollständige Syntax können Sie der Beschreibung des Befehls **shutdown** in der Veröffentlichung "Commands Reference" entnehmen.

### **Einen anderen Befehl oder ein anderes Programm suchen (Befehl whereis):**

Mit dem Befehl **whereis** können die Quellen, Binärdaten und Man-Pages zu bestimmten Dateien gesucht werden. Nach Eingabe des Befehls wird das gesuchte Programm in einer Liste von Standardverzeichnis gesucht.

Beispiele:

- Geben Sie Folgendes ein, um Dateien im aktuellen Verzeichnis zu suchen, zu denen es keine Dokumentation gibt:

```
whereis -m -u *
```

- Geben Sie Folgendes ein, um alle Dateien zu suchen, die den Namen Mail enthalten:

```
whereis Mail
```

Das System zeigt Informationen ähnlich den folgenden an:

```
Mail: /usr/bin/Mail /usr/lib/Mail.rc
```

Die vollständige Syntax des Befehls **whereis** ist in der Veröffentlichung *Commands Reference, Volume 6* beschrieben.

### **Informationen zu einem Befehl anzeigen (Befehl man):**

Mit dem Befehl **man** werden Informationen zu Befehlen, Subroutinen und Dateien angezeigt.

Das Standardformat für den Befehl **man** ist folgendermaßen:

```
man Befehlsname
```



Geben Sie Folgendes ein, um Informationen zum Befehl **pg** anzuzeigen:

```
man pg
```

Das System zeigt Informationen ähnlich den folgenden an:

Befehl **pg**

Zweck

Formatiert die Dateien für die Anzeige am Bildschirm.

Syntax

```
pg [- Nummer] [-c] [-e] [-f] [-n] [-p Zeichenfolge]
[-s] [+Zeilennummer | +/Muster/] [Datei ...]
```

Beschreibung

Der Befehl **pg** liest die mit dem Parameter "Datei" angegebene Datei und schreibt diese Datei seitenweise in die Standardausgabe. Wenn Sie für den Parameter "Datei" ein Minuszeichen (-) angeben oder den Befehl **pg** ohne Optionen ausführen, liest der Befehl **pg** die Standardeingabe. Am Ende jeder angezeigten wird eine weitere Seite angezeigt. Mit den Unterbefehlen für den Befehl **pg** können Sie die Datei anzeigen und in der Datei suchen.

Die vollständige Syntax des Befehls **man** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

#### **Funktion eines Befehls anzeigen (Befehl **whatis**):**

Der Befehl **whatis** schlägt einen bestimmten Befehl, Systemaufruf, eine Bibliotheksfunktion oder einen speziellen Dateinamen, der mit dem Parameter **Befehl** angegeben wird, in einer mit dem Befehl **catman -w** erstellten Datenbank nach.

Weitere Informationen finden Sie in der Beschreibung des Befehls **catman -w** in der Veröffentlichung "Commands Reference". Anschließend zeigt der Befehl **whatis** die Kopfzeile aus dem Man-Abschnitt an. Sie können anschließend den Befehl **man** eingeben, um zusätzliche Informationen anzuzeigen. Weitere Informationen finden Sie in der Beschreibung des Befehls **man** in der Veröffentlichung "Commands Reference".

Der Befehl **whatis** entspricht dem Befehl **man -f**.

Geben Sie Folgendes ein, um herauszufinden, welche Funktion der Befehl **ls** hat:

```
whatis ls
```

Das System zeigt Informationen ähnlich den folgenden an:

```
ls(1) - Zeigt den Inhalt eines Verzeichnisses an.
```

Die vollständige Syntax des Befehls **whatis** ist in der Veröffentlichung *Commands Reference, Volume 6* beschrieben.

#### **Zuvor eingegebene Befehle auflisten (Befehl **history**):**

Mit dem Befehl **history** können Sie die Befehle auflisten, die Sie zuvor eingegeben haben.

Der Befehl **history** ist ein integrierter Korn-Shellbefehl, der die 16 zuletzt eingegebenen Befehle auflistet. Die Korn-Shell speichert die eingegebenen Befehle in einer Befehlsprotokolldatei, die gewöhnlich den Namen `$HOME/.sh_history` hat. Wenn Sie einen bereits eingegebenen Befehl wiederholen müssen, können Sie mit diesem Befehl Zeit einsparen.

Standardmäßig speichert die Korn-Shell den Text der letzten 128 Befehle für andere Benutzer als Root und die letzten 512 Befehle für Root. Die Größe der Protokolldatei (Umgebungsvariable *HISTSIZE*) ist nicht begrenzt. Es ist jedoch zu beachten, dass eine große Protokolldatei die Initialisierung der Korn-Shell verlangsamen kann.

**Anmerkung:** Die Bourne-Shell unterstützt keine Befehlsprotokolle.

Geben Sie an der Eingabeaufforderung Folgendes ein, um die zuvor eingegebenen Befehle aufzulisten:

```
history
```

Der Befehl **history**, ohne Optionen eingegeben, listet die 16 zuletzt eingegebenen Befehle auf. Das System zeigt Informationen ähnlich den folgenden an:

```
928 ls
929 mail
930 printenv MAILMSG
931 whereis Mail
932 whatis ls
933 cd /usr/include/sys
934 ls
935 man pg
936 cd
937 ls | pg
938 lscons
939 tty
940 ls *.txt
941 printenv MAILMSG
942 pwd
943 history
```

In der ersten Spalte der Liste wird die Position des Befehls in der Datei `$HOME/.sh_history` angezeigt, in der zweiten Spalte der Befehl selbst.

Geben Sie an der Eingabeaufforderung Folgendes ein, um die fünf zuletzt eingegebenen Befehle anzuzeigen:

```
history -5
```

Es wird eine Liste ähnlich der folgenden angezeigt:

```
939 tty
940 ls *.txt
941 printenv MAILMSG
942 pwd
943 history
944 history -5
```

Wenn Sie mit dem Befehl **history** eine Zahl angeben, werden alle zuvor ausgeführten Befehle ab der angegebenen Nummer aufgelistet.

Geben Sie an der Eingabeaufforderung Folgendes ein, um die Befehle ab Nummer 938 aufzulisten:

```
history 938
```

Es wird eine Liste ähnlich der folgenden angezeigt:

```
938 lscons
939 tty
940 ls *.txt
941 printenv MAILMSG
942 pwd
943 history
944 history -5
945 history 938
```

### Zugehörige Konzepte:

„Betriebssystem-Shells“ auf Seite 217

Die Schnittstelle zum Betriebssystem wird als *Shell* bezeichnet.

„Befehlsprotokollsubstitution“ auf Seite 269

Mit dem integrierten Befehl `fc` können Teile der Protokolldatei aufgelistet oder editiert werden. Wenn Sie einen Teil der Datei zur Bearbeitung bzw. Auflistung auswählen möchten, geben Sie die Nummer oder das erste bzw. die ersten Zeichen des Befehls ein.

### Befehle mit dem Aliasbefehl `r` wiederholen:

Verwenden Sie den Korn-Shellalias `r`, um vorherige Befehle zu wiederholen.

Geben Sie `r` ein, und drücken Sie anschließend die Eingabetaste. Sie können auch die Nummer oder das bzw. die ersten Zeichen des Befehls eingeben.

Wenn Sie die auf dem System verfügbaren Bildschirme auflisten möchten, geben Sie an der Eingabeaufforderung den Befehl `lsdisp` ein. Das System gibt die Informationen am Bildschirm aus. Sollen dieselben Informationen erneut angezeigt werden, geben Sie an der Eingabeaufforderung Folgendes ein:

```
r
```

Das System führt den letzten Befehl erneut aus. In diesem Beispiel wird der Befehl `lsdisp` ausgeführt.

Geben Sie an der Eingabeaufforderung Folgendes ein, um den Befehl `ls *.txt` zu wiederholen:

```
r ls
```

Mit dem Korn-Shellalias `r` wird der letzte Befehl gesucht, der mit den angegebenen Zeichen beginnt.

### Zeichenfolgesubstitution mit dem Aliasbefehl `r`:

Mit dem Korn-Shellaliasbefehl `r` können Sie einen Befehl ändern, bevor Sie ihn ausführen.

In diesem Fall kann ein Substitutionsparameter im Format *Alt=Neu* verwendet werden, um den Befehl vor der Ausführung zu ändern.

Die folgenden Beispiele veranschaulichen die Verwendung des Aliasnamens `r`:

- Wenn die Befehlszeile 940 `ls *.txt` lautet und Sie an der Eingabeaufforderung den Befehl `ls *.exe` ausführen möchten, geben Sie Folgendes ein:

```
r txt=exe 940
```

Mit diesem Befehl wird der Befehl 940 ausgeführt, wobei `txt` durch `exe` ersetzt wird.

- Wenn der Befehl in Zeile 940 der zuletzt ausgeführte Befehl ist, der mit dem Kleinbuchstaben `l` beginnt, können Sie auch Folgendes eingeben:

```
r txt=exe l
```

**Anmerkung:** Es wird jeweils nur das erste Vorkommen der Zeichenfolge *Alt* durch die Zeichenfolge *Neu* ersetzt. Wenn Sie den Korn-Shellalias `r` ohne Angabe einer Befehlsnummer oder eines Zeichens eingeben, wird die Substitution für den zuletzt ausgeführten Befehl durchgeführt.

### Befehlsprotokoll bearbeiten:

Verwenden Sie den integrierten Korn-Shellbefehl `fc`, um Teile der Befehlsprotokolldatei aufzulisten oder zu bearbeiten.

Wenn Sie einen Teil der Datei zur Bearbeitung bzw. Auflistung auswählen möchten, geben Sie die Nummer oder das erste bzw. die ersten Zeichen des Befehls ein. Sie können einen einzelnen Befehl oder einen Bereich von Befehlen angeben.

Wenn Sie mit dem integrierten Korn-Shellbefehl **fc** kein Editorprogramm als Argument angeben, wird der Editor verwendet, der in der Variablen *FCEDIT* definiert ist. Ist die Variable *FCEDIT* nicht definiert, wird der Editor `/usr/bin/ed` verwendet. Die editierten Befehle werden beim Verlassen des Editors ausgegeben und ausgeführt. Mit dem Befehl **printenv** kann der Wert der Variablen *FCEDIT* angezeigt werden.

Die folgenden Beispiele veranschaulichen, wie das Befehlsprotokoll bearbeitet werden kann:

- Wenn Sie den Befehl

```
cd /usr/tmp
```

ausführen möchten, der der Befehlszeile 933 gleicht, geben Sie an der Eingabeaufforderung Folgendes ein:

```
fc 933
```

Daraufhin wird in Ihrem Standardeditor die Befehlszeile 933 angezeigt. Ändern Sie `include/sys` in `tmp`. Wenn Sie den Editor verlassen, wird der bearbeitete Befehl ausgeführt.

- Sie können mit dem Befehl **fc** auch einen anderen zu verwendenden Editor angeben. Wenn Sie einen Befehl beispielsweise mit dem Editor `/usr/bin/vi` bearbeiten möchten, müssen Sie an der Eingabeaufforderung Folgendes eingeben:

```
fc -e vi 933
```

Daraufhin wird im Editor `vi` die Befehlszeile 933 angezeigt.

- Sie können auch einen Bereich von zu bearbeitenden Befehlen angeben. Geben Sie an der Eingabeaufforderung Folgendes ein, um beispielsweise die Befehle 930 bis 940 zu bearbeiten:

```
fc 930 940
```

Daraufhin wird der Standardeditor aufgerufen, in dem die Befehle 930 bis 940 angezeigt werden. Beim Verlassen des Editors werden alle im Editor angezeigten Befehle nacheinander ausgeführt.

### **Befehlsalias erstellen (Shellbefehl alias):**

Mit einem *Aliasnamen* können Sie Namen für den Direktaufruf von Befehlen, Dateien und Shellprozeduren erstellen. Durch die Verwendung von Aliasnamen können Sie Zeit einsparen, wenn bestimmte Tasks häufig ausgeführt werden müssen. Sie können einen Befehlsalias erstellen.

Mit dem integrierten Korn-Shellbefehl **alias** können Sie ein Wort als Aliasnamen für einen Befehl definieren. Sie können Aliasnamen verwenden, um bestimmte integrierte Befehle erneut zu definieren. Mit Aliasnamen können jedoch keine reservierten Maschinenwörter erneut definiert werden.

Das erste Zeichen eines Aliasnamens kann jedes druckbare Zeichen außer einem Metazeichen sein. Für die verbleibenden Zeichen gelten dieselben Regeln wie für gültige Dateinamen.

Zum Erstellen eines Aliasnamens wird folgendes Format verwendet:

```
alias Name=Zeichenfolge
```

Der Parameter *Name* gibt den Aliasnamen und der Parameter *Zeichenfolge* eine Zeichenfolge an. Wenn die *Zeichenfolge* Leerzeichen enthält, schließen Sie diese in Anführungszeichen ein.

Die folgenden Beispiele veranschaulichen, wie ein Aliasname erstellt wird:

- Geben Sie zum Erstellen eines Aliasnamens für den Befehl **rm -i** (Anfrage vor dem Löschen von Dateien) an der Eingabeaufforderung Folgendes ein:

```
alias rm="/usr/bin/rm -i"
```

Wenn Sie danach den Befehl **rm** eingeben, wird eigentlich der Befehl `/usr/bin/rm -i` ausgeführt.

- Wenn Sie den Aliasnamen **dir** für den Befehl `ls -a1F | pg` (der detaillierte Informationen zu allen Dateien im aktuellen Verzeichnis, einschließlich der nicht sichtbaren Dateien, anzeigt, die ausführbaren Dateien mit einem Stern (\*) und Verzeichnisse mit einem Schrägstrich (/) kennzeichnet und die Liste seitenweise anzeigt) erstellen möchten, geben Sie an der Eingabeaufforderung Folgendes ein:

```
alias dir="/usr/bin/ls -a1F | pg"
```

Wenn Sie danach den Befehl **dir** eingeben, wird eigentlich der Befehl `/usr/bin/ls -a1F | pg` ausgeführt.

- Geben Sie an der Eingabeaufforderung Folgendes ein, um alle vorhandenen Aliasnamen anzuzeigen:  
alias

Das System zeigt Informationen ähnlich den folgenden an:

```
rm="/usr/bin/rm -i"
dir="/usr/bin/ls -a1F | pg"
```

### Zugehörige Konzepte:

„Befehls-Aliasing in der Korn- bzw. POSIX-Shell“ auf Seite 270

In der Korn- bzw. POSIX-Shell können Sie Aliasnamen für die Anpassung von Befehlen erstellen.

### Unterstützung internationaler Zeichensätze bei der Textformatierung:

Sie können die Textformatierungsbefehle verwenden, um Text zu bearbeiten, der aus dem für die europäischen Sprachen benutzten internationalen erweiterten Zeichensatz stammt.

Der internationale erweiterte Zeichensatz stellt die in vielen europäischen Sprachen benutzten Zeichen und Symbole sowie einen ASCII-Teilsatz zur Verfügung, der aus den Buchstaben, Ziffern und Interpunktionszeichen der englischen Sprache besteht.

Alle Zeichen des europäischen erweiterten Zeichensatzes haben das ASCII-Format. Dieses Format kann für die Darstellung erweiterter Zeichen bei der Eingabe benutzt werden. Die Zeichen können auch direkt über eine Einheit, z. B. über eine Tastatur, die europäische erweiterte Zeichen unterstützt, eingegeben werden.

Die nachfolgend aufgeführten Textformatierungsbefehle können für die Bearbeitung von Text in verschiedenen Sprachen benutzt werden, sofern es sich bei den länderspezifischen Zeichen um Einzelbytezeichen handelt. Sie finden diese Befehle in `/usr/bin`. (Die mit einem Stern (\*) gekennzeichneten Befehle unterstützen Mehrbytesprachen.

addbib*	hyphen	pic*	pstext
checkmm	ibm3812	ps4014	refer*
checknr*	ibm3816	ps630	roffbib*
col*	ibm5587G*	psbanne	soelim*
colcrt	ibm5585H-T*	psdit	sortbib*
deroff*	indxbib*	psplot	tbl*
enscript	lookbib*	psrev	troff*
eqn*	makedev*	psroff	vgrind
grap*	neqn*	psrv	xpreview*
hplj	nroff*		

Textformatierungsbefehle und Makropakete, die nicht in dieser Liste enthalten sind, können nicht für die Bearbeitung von internationalen Zeichen benutzt werden.

### Zugehörige Konzepte:

„Unterstützung von Mehrbytezeichen bei der Textformatierung“ auf Seite 142

Einige Textformatierungsbefehle können zur Bearbeitung von Text in Mehrbytesprachen verwendet werden.

### Textformatierung mit erweiterten Einzelbytezeichen:

Unterstützt die Eingabeeinheit die Zeichen des erweiterten Zeichensatzes für europäische Sprachen, können diese Zeichen direkt eingegeben werden.

Andernfalls können die Zeichen mithilfe einer ASCII-Escape-Zeichenfolge in folgendem Format dargestellt werden:

`\[N]`, wobei *N* für den 2- oder 4-stelligen Hexadezimalcode des Zeichens steht.

**Anmerkung:** Das NCesc-Format `\<xx>` wird nicht mehr unterstützt.

Text, der erweiterte Zeichen enthält, wird gemäß den Formatierungskonventionen der jeweiligen Sprache ausgegeben. Zeichen, die nicht für die Schnittstelle zu einer bestimmten Ausgabereinheit definiert sind, werden nicht ausgegeben oder verursachen ein Fehlersymptom.

Obwohl die Namen der Anforderungen, Makropakete und Befehle auf der englischen Sprache basieren, werden Eingaben (wie Dateinamen und Parameter), die Zeichen des europäischen erweiterten Zeichensatzes enthalten, normalerweise akzeptiert.

Für die Befehle **nroff** und **troff** sowie die zugehörigen Vorprozessoren müssen ASCII-Zeichen eingegeben werden. Andernfalls tritt ein nicht behebbarer Syntaxfehler auf. Zeichen des internationalen Zeichensatzes (Einzel- oder Mehrbytezeichen) können in Anführungszeichen eingeschlossen oder in anderem zu formatierenden Text eingegeben werden. Beispiel für die Verwendung von Makros im Befehl **pic**:

```
define foobar % EinText %
```

Hinter der Anweisung `define` muss der angegebene Name `foobar` in ASCII eingegeben werden. Der Ersatztext, `Text`, kann jedoch auch Zeichen enthalten, die keine ASCII-Zeichen sind.

### Unterstützung von Mehrbytezeichen bei der Textformatierung:

Einige Textformatierungsbefehle können zur Bearbeitung von Text in Mehrbytesprachen verwendet werden.

Diese Befehle sind in der Liste unter "Unterstützung nationaler Zeichensätze bei der Textformatierung" mit einem Stern (\*) gekennzeichnet. Textformatierungsbefehle, die nicht in der Liste enthalten sind, können nicht für die Bearbeitung von internationalen Zeichen benutzt werden.

Mehrbytezeichen können direkt eingegeben werden, falls dieses Verfahren von der Eingabeeinheit unterstützt wird. Andernfalls können Mehrbytezeichen im ASCII-Format `\[N]` eingegeben werden, wobei *N* der 2-, 4-, 6-, 7- oder 8-stellige Hexadezimalcode für das Zeichen ist.

Obwohl die Namen der Anforderungen, Makropakete und Befehle auf der englischen Sprache basieren, werden Eingaben (wie Dateinamen und Parameter), die Zeichen des europäischen erweiterten Zeichensatzes enthalten, normalerweise akzeptiert.

In der folgenden Liste sind die Merkmale zusammengefasst, die gegenüber den Textformatierungsbefehlen für Einzelbytetext hervorzuheben sind oder bei denen es sich um spezifische Merkmale für Mehrbytezeichensätze handelt.

- Keine Silbentrennung im Text.
- Für die Ausgabe numerischer Mehrbytezeichen werden spezielle Formattypen benötigt. Die Formattypen für Japanisch sind verfügbar.
- Der Text wird in Zeilen von links nach rechts ausgegeben.
- Der Zeichenabstand bleibt konstant, so dass die Zeichen automatisch in Spalten ausgerichtet werden.

- Zeichen, die nicht für die Schnittstelle zu einer bestimmten Ausgabeinheit definiert sind, werden nicht ausgegeben oder verursachen ein Fehlersymptom.

### Zugehörige Konzepte:

„Unterstützung internationaler Zeichensätze bei der Textformatierung“ auf Seite 141

Sie können die Textformatierungsbefehle verwenden, um Text zu bearbeiten, der aus dem für die europäischen Sprachen benutzten internationalen erweiterten Zeichensatz stammt.

### Kalender anzeigen:

Mit dem Befehl **cal** können Sie einen Kalender in die Standardausgabe schreiben.

Mit dem Parameter **Monat** wird der Monat für den Kalender angegeben. Es kann eine Zahl von 1 bis 12 für Januar bis Dezember angegeben werden. Wird der Parameter **Monat** nicht angegeben, verwendet der Befehl **cal** standardmäßig den aktuellen Monat.

Mit dem Parameter **Jahr** wird das Jahr für den Kalender angegeben. Da der Befehl **cal** einen Kalender für ein beliebiges Jahr von 1 bis 9999 anzeigen kann, geben Sie die vollständige Jahreszahl an und nicht nur die letzten beiden Ziffern. Wird der Parameter **Jahr** nicht angegeben, verwendet der Befehl **cal** standardmäßig das aktuelle Jahr.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **cal**:

1. Wenn Sie einen Kalender für Februar 2002 auf Ihrer Workstation anzeigen möchten, geben Sie Folgendes ein:

```
cal 2 2002
```

2. Drücken Sie die Eingabetaste.

3. Wenn Sie einen Kalender für das Jahr 2002 drucken möchten, geben Sie Folgendes ein:

```
cal 2002 | qprt
```

4. Drücken Sie die Eingabetaste.

Die vollständige Syntax finden Sie in der Beschreibung des Befehls **cal** in der Veröffentlichung *Commands Reference, Volume 1*.

### Erinnerungsnachrichten anzeigen:

Sie können eine Erinnerungsnachricht anzeigen, indem Sie eine Datei namens **calendar** einlesen. Diese Datei wird in Ihrem Ausgangsverzeichnis mit dem Befehl **calendar** erstellt. Der Befehl schreibt jede Zeile der Datei mit dem Datum von heute oder morgen in die Standardausgabe.

Sie können eine Datei mit dem Namen **calendar** einlesen, die Sie mit dem Befehl **calendar** in Ihrem Ausgangsverzeichnis erstellt haben. Der Befehl schreibt jede Zeile der Datei mit dem Datum von heute oder morgen in die Standardausgabe.

Der Befehl **calendar** erkennt Datumsformate wie Dec. 7 bzw. 12/7. Er erkennt auch das Sonderzeichen Stern (\*), wenn anschließend ein Schrägstrich (/) folgt. Er interpretiert beispielsweise \*/7 als siebten Tag jedes Monats.

An Freitagen gibt der Befehl **calendar** alle Termine für die Tage Freitag, Samstag, Sonntag und Montag aus. Der Befehl erkennt jedoch keine Feiertage. An Feiertagen funktioniert der Befehl wie gehabt und gibt den Terminplan für den nächsten Tag aus.

### Eine typische Kalenderdatei verwenden

Eine Datei vom Typ **calendar** könnte beispielsweise so aussehen:

\*/25 - Prepare monthly report  
Aug. 12 - Fly to Denver  
aug 23 - board meeting  
Martha out of town - 8/23, 8/24, 8/25  
8/24 - Mail car payment  
sat aug/25 - beach trip  
August 27 - Meet with Simmons  
August 28 - Meet with Wilson

Geben Sie Folgendes ein, um den Befehl **calendar** auszuführen:

```
calendar
```

Wenn heute Freitag, der 24. August, ist, zeigt der Befehl **calendar** Folgendes an:

```
*/25 - Prepare monthly report
Martha out of town - 8/23, 8/24, 8/25
8/24 - Mail car payment
sat aug/25 - beach trip
August 27 - Meet with Simmons
```

### Kalenderdatei mit einer include-Anweisung verwenden

Eine Datei **calendar** mit einer **include**-Anweisung könnte beispielsweise wie folgt aussehen:

```
#include </tmp/out>
1/21 -Annual review
1/21 -Weekly project meeting
1/22 *Meet with Harrison in Dallas*
Doctor's appointment - 1/23
1/23 -Vinh's wedding
```

Geben Sie Folgendes ein, um den Befehl **calendar** auszuführen:

```
calendar
```

Wenn heute Mittwoch, der 21. Januar, ist, zeigt der Befehl **calendar** Folgendes an:

```
Jan.21 Goodbye party for David
Jan.22 Stockholder meeting in New York
1/21 -Annual review
1/21 -Weekly project meeting
1/22 *Meet with Harrison in Dallas*
```

Aus den angezeigten Ergebnissen des Befehls **calendar** geht hervor, dass die Datei **/tmp/out** folgende Zeilen enthält:

```
Jan.21 Goodbye party for David
Jan.22 Stockholder meeting in New York
```

Die vollständige Syntax finden Sie in der Beschreibung des Befehls **calendar** in der Veröffentlichung *Commands Reference, Volume 1*.

### Zahl in Primfaktoren zerlegen:

Sie können Zahlen mit dem Befehl **factor** in Primfaktoren zerlegen.

Wenn Sie den Befehl **factor** absetzen, ohne einen Wert für den Parameter **Nummer** anzugeben, wartet der Befehl auf die Eingabe einer positiven Zahl, die kleiner ist als 1E14 (100.000.000.000.000). Er schreibt dann anschließend den Primfaktor dieser Zahl in die Standardausgabe. Er zeigt die Primfaktoren sortiert und die Anzahl der Vorkommen an, sofern derselbe Faktor mehr als einmal verwendet wird. Geben Sie zum Beenden die Zahl 0 (Null) oder ein nicht numerisches Zeichen ein.

Wenn Sie den Befehl **factor** mit einem Argument absetzen, bestimmt der Befehl die Primfaktoren des Parameters **Nummer**, schreibt das Ergebnis in die Standardausgabe und endet.



Im Folgenden sehen Sie ein Beispiel für die Berechnung von Primfaktoren:

1. Geben Sie Folgendes ein, um die Primfaktoren der Zahl 123 zu berechnen:

```
factor 123
```

2. Drücken Sie die Eingabetaste. Es wird Folgendes angezeigt:

```
123 3 41
```

Die vollständige Syntax finden Sie in der Beschreibung des Befehls **factor** in der Veröffentlichung *Commands Reference, Volume 2*.

### **Befehl mit Schlüsselwort suchen:**

Sie können Man-Page-Abschnitte anzeigen, die beliebige der angegebenen *Schlüsselwörter* in ihrem Titel enthalten. Verwenden Sie hierfür den Befehl **apropos**.

Der Befehl **apropos** behandelt jedes Wort gesondert und beachtet nicht die Groß-/Kleinschreibung. Wörter, die Teil anderer Wörter sind, werden auch angezeigt. Wenn Sie beispielsweise nach dem Wort *compile* suchen, findet der Befehl **apropos** auch alle Vorkommen des Worts *compiler*.

**Anmerkung:** Die Datenbank mit den Schlüsselwörtern befindet sich im Verzeichnis `/usr/share/man/whatis` und muss zuerst mit dem Befehl **catman -w** generiert werden.

Der Befehl **apropos** entspricht der Verwendung des Befehls **man** mit der Option **-k**.

Wenn Sie beispielsweise Handbuchabschnitte finden möchten, die das Wort *password* in der Überschrift enthalten, geben Sie Folgendes ein:

```
apropos password
```

Drücken Sie die Eingabetaste.

Die vollständige Syntax des Befehls **apropos** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

### **Prozesse**

Auf einem Computer ausgeführte Programme und Befehle werden als *Prozesse* bezeichnet.

Es gibt Eltern- und Kindprozesse. Ein Prozess, der von einem Programm oder Befehl gestartet wird, ist ein *Elternprozess*. Ein *Kindprozess* ist das Produkt eines Elternprozesses. Ein Elternprozess kann mehrere Kindprozesse haben. Ein Kindprozess kann jedoch nur einen einzigen Elternprozess haben.

Jedem Prozess wird vom System beim Starten eine Prozessidentifikationsnummer (Prozess-ID) zugeordnet. Wird dasselbe Programm mehrmals gestartet, erhält es jedes Mal eine andere Prozess-ID.

Jeder Prozess, der auf einem System gestartet wird, nutzt einen Teil der verfügbaren Systemressourcen. Werden mehrere Prozesse gleichzeitig ausgeführt, teilt ein im Betriebssystem integrierter Scheduler jedem Prozess auf der Basis festgelegter Prioritäten einen Anteil an Rechenzeit zu. Diese Prioritäten können mit den Befehlen **nice** und **renice** geändert werden.

**Anmerkung:** Zum Zuweisen einer höheren Prozesspriorität müssen Sie Rootberechtigung besitzen. Alle Benutzer können mit dem Befehl **nice** einem Prozess, den sie starten, und mit dem Befehl **renice** einem Prozess, den sie bereits gestartet haben, eine niedrigere Priorität zuweisen.

In der folgenden Liste sind die verschiedenen Prozesstypen beschrieben:

### **Vordergrund- und Hintergrundprozesse**

Prozesse, die von einem Benutzer gestartet werden oder mit dem Benutzer interagieren müssen, werden als *Vordergrundprozesse* bezeichnet. Prozesse, die ohne Benutzerintervention ausgeführt werden, bezeichnet man als *Hintergrundprozesse*. Programme und Befehle werden standardmäßig als Vordergrundprozesse ausgeführt. Wenn Sie einen Prozess im Hintergrund ausführen möchten, geben Sie am Ende des Befehls, mit dem der Prozess gestartet wird, ein Et-Zeichen (&) ein.

### Dämonprozesse

*Dämonprozesse* sind unabhängige Prozesse, auf die der Benutzer keinen Einfluss hat. Sie werden ständig im Hintergrund ausgeführt und sind jederzeit verfügbar. Dämonprozesse werden normalerweise beim Starten des Systems gestartet und so lange ausgeführt, bis das System ausgeschaltet wird. Sie werden in der Regel zur Ausführung von Systemservices verwendet und stehen jederzeit mehreren Tasks und Benutzern zur Verfügung. Dämonprozesse werden von Root oder von der Root-Shell gestartet und können nur von Root wieder beendet werden. Beispielsweise wird vom Prozess **qdaemon** der Zugriff auf Systemressourcen, wie z. B. Drucker, hergestellt. Ein weiterer allgemeiner Dämonprozess ist der Dämonprozess **sendmail**.

### Geisterprozesse

Bei einem *Geisterprozess* (oder *Zombie-Prozess*) handelt es sich um einen inaktiven Prozess, der in der Prozesstabelle zwar noch geführt wird (d. h., ihm ist immer noch eine Prozess-ID zugeordnet), dem aber kein Systemspeicherbereich mehr zugeordnet ist. Geisterprozesse sind beendete Prozesse, die in der Prozesstabelle so lange weiter existieren, bis der zugehörige Elternprozess beendet oder ein Systemabschluss durchgeführt und das System erneut gestartet wird. Geisterprozesse haben den Status <fehlerhaft>, wenn sie mit dem Befehl **ps** aufgelistet werden.

### Prozessstart:

Vordergrundprozesse werden von einem Datensichtgerät aus durch Eingabe eines Programm- oder Befehlsnamens an der Eingabeaufforderung gestartet.

Nach dem Starten eines Vordergrundprozesses finden während der Verarbeitungszeit Dialoge zwischen dem Benutzer am Datensichtgerät und dem Prozess statt. Andere Interaktionen (z. B. die Eingabe eines anderen Befehls) können erst dann wieder ausgeführt werden, wenn der Prozess beendet ist oder vom Benutzer gestoppt wird.

Ein einzelner Benutzer kann standardmäßig bis zu 40 Prozesse gleichzeitig ausführen.

### Einen Prozess im Vordergrund starten

Zum Starten von Prozessen im Vordergrund geben Sie den Namen des Befehls mit den gültigen Parametern und Flags ein:

```
$ Befehlsname
```

### Einen Prozess im Hintergrund starten

Geben Sie zum Ausführen eines Prozesses im Hintergrund den Namen des Befehls mit den erforderlichen Parametern und Flags ein, und schließen Sie den Befehl mit einem Et-Zeichen (&) ab:

```
$ Befehlsname&
```

Während der Ausführung eines Hintergrundprozesses können weitere Tasks durch Eingabe der entsprechenden Befehle am Datensichtgerät ausgeführt werden.

Im Allgemeinen eignen sich Hintergrundprozesse für Befehle mit langer Ausführungsdauer. Da Hintergrundprozesse jedoch die Gesamtarbeitslast des Prozessors erhöhen, können sie sich auf die Gesamtleistung des Systems auswirken.

Die meisten Prozesse, auch Hintergrundprozesse, leiten ihre Ausgabe an eine Standardausgabeeinheit. Die Standardausgabe wird an den Bildschirm geleitet, sofern keine Umleitung definiert ist. Um unerwünschte Arbeitsunterbrechungen wegen der Ausgabe von Hintergrundprozessen zu

vermeiden, empfiehlt es sich, die Ausgabe von Hintergrundprozessen in eine Datei oder an einen Drucker umzuleiten. Dort ist die Ausgabe jederzeit verfügbar.

**Anmerkung:** Unter bestimmten Umständen kann ein Prozess im Hintergrund seine Ausgabe in einer anderen Reihenfolge als im Vordergrund generieren. Mit der Subroutine **fflush** können Programmierer sicherstellen, dass die Ausgabe immer in der richtigen Reihenfolge generiert wird, unabhängig davon, ob der Prozess im Vordergrund oder im Hintergrund ausgeführt wird.

Der Status eines Hintergrundprozesses kann mit dem Befehl **ps** überprüft werden.

### Befehl zum Überprüfen des Prozessstatus (Befehl **ps**):

Während des Systembetriebs werden auch Prozesse ausgeführt. Mit dem Befehl **ps** können Sie ermitteln, welche Prozesse aktiv sind, und Informationen zu diesen Prozessen anzeigen.

Der Befehl **ps** besitzt mehrere Flags, mit denen angegeben werden kann, welche Prozesse aufgelistet und welche Informationen zu den einzelnen Prozessen angezeigt werden sollen.

Geben Sie Folgendes ein, um alle auf dem System ausgeführten Prozesse anzuzeigen:

```
ps -ef
```

Das System zeigt Informationen ähnlich den folgenden an:

```
USER PID PPID C STIME TTY TIME CMD
root 1 0 0 Jun 28 - 3:23 /etc/init
root 1588 6963 0 Jun 28 - 0:02 /usr/etc/biod 6
root 2280 1 0 Jun 28 - 1:39 /etc/syncd 60
mary 2413 16998 2 07:57:30 - 0:05 aixterm
mary 11632 16998 0 07:57:31 lft/1 0:01 xbiff
mary 16260 2413 1 07:57:35 pts/1 0:00 /bin/ksh
mary 16469 1 0 07:57:12 lft/1 0:00 ksh /usr/lpp/X11/bin/xinit
mary 19402 16260 20 09:37:21 pts/1 0:00 ps -ef
```

Die Spalten in der vorherigen Ausgabe sind wie folgt definiert:

Eintrag	Beschreibung
USER	Anmeldename des Benutzers
PID	Prozess-ID
PPID	Prozess-ID des Elternprozesses
C	CPU (Auslastung durch den Prozess)
STIME	Startzeit des Prozesses
TTY	Workstationsteuerung für den Prozess
TIME	Gesamte Ausführungszeit für den Prozess
CMD	Befehl

Die Prozess-ID für den Befehl **ps -ef** im vorherigen Beispiel ist 19402. Die Prozess-ID des Elternprozesses (Befehl **/bin/ksh**) ist 16260.

Falls die Liste sehr umfangreich ist, verschiebt sich der obere Teil unter Umständen aus dem Sichtbereich. Wenn Sie die Liste seitenweise anzeigen möchten, müssen Sie den Befehl **ps** durch ein Pipe-Symbol mit dem Befehl **pg** verketteten. Geben Sie an der Eingabeaufforderung Folgendes ein:

```
ps -ef | pg
```

Geben Sie Folgendes an der Eingabeaufforderung ein, um Statusinformationen zu allen auf dem System ausgeführten Prozessen anzuzeigen:

```
ps gv
```

Mit diesem Befehl wird eine Reihe von Statistiken für jeden aktiven Prozess aufgelistet. Die Ausgabe des Befehls sieht in etwa wie folgt aus:

PID	TTY	STAT	TIME	PGIN	SIZE	RSS	LIM	TSIZ	TRS	%CPU	%MEM	COMMAND
0	-	A	0:44	7	8	8	xx	0	0	0.0	0.0	swapper
1	-	A	1:29	518	244	140	xx	21	24	0.1	1.0	/etc/init
771	-	A	1:22	0	16	16	xx	0	0	0.0	0.0	kproc
1028	-	A	0:00	10	16	8	xx	0	0	0.0	0.0	kproc
1503	-	A	0:33	127	16	8	xx	0	0	0.0	0.0	kproc
1679	-	A	1:03	282	192	12	32768	130	0	0.7	0.0	pcidossvr
2089	-	A	0:22	918	72	28	xx	1	4	0.0	0.0	/etc/sync
2784	-	A	0:00	9	16	8	xx	0	0	0.0	0.0	kproc
2816	-	A	5:59	6436	2664	616	8	852	156	0.4	4.0	/usr/lpp/
3115	-	A	0:27	955	264	128	xx	39	36	0.0	1.0	/usr/lib/
3451	-	A	0:00	0	16	8	xx	0	0	0.0	0.0	kproc
3812	-	A	0:00	21	128	12	32768	34	0	0.0	0.0	usr/lib/lpd/
3970	-	A	0:00	0	16	8	xx	0	0	0.0	0.0	kproc
4267	-	A	0:01	169	132	72	32768	16	16	0.0	0.0	/etc/sysl
4514	lft/0	A	0:00	60	200	72	xx	39	60	0.0	0.0	/etc/gett
4776	pts/3	A	0:02	250	108	280	8	303	268	0.0	2.0	-ksh
5050	-	A	0:09	1200	424	132	32768	243	56	0.0	1.0	/usr/sbin
5322	-	A	0:27	1299	156	192	xx	24	24	0.0	1.0	/etc/cron
5590	-	A	0:00	2	100	12	32768	11	0	0.0	0.0	/etc/writ
5749	-	A	0:00	0	208	12	xx	13	0	0.0	0.0	/usr/lpp/
6111	-	T	0:00	66	108	12	32768	47	0	0.0	0.0	/usr/lpp/

Die vollständige Syntax des Befehls **ps** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

#### Anfangspriorität eines Prozesses festlegen (Befehl nice):

Sie können für einen Prozess eine Anfangspriorität festlegen, die niedriger ist als die Basispriorität.

Sie können für einen Prozess eine Anfangspriorität festlegen, die niedriger ist als die Basispriorität. Geben Sie den Befehl **nice** ein, um den Prozess zu starten.

**Anmerkung:** Wenn Sie einen Prozess mit einer höheren Priorität als der Basisausführungspriorität ausführen möchten, müssen Sie Rootberechtigung besitzen.

Geben Sie zum Festlegen der Anfangspriorität für einen Prozess Folgendes ein:

```
nice -n Zahl Befehlsfolge
```

*Zahl* steht für den Bereich von 0 bis 39, wobei 39 die niedrigste Priorität ist. Der *nice-Wert* ist der Dezimalwert der Systemplanungspriorität eines Prozesses. Je höher die *Zahl* ist, desto niedriger ist die Priorität. Wenn Sie die Priorität null verwenden, wird der Prozess mit der Basispriorität ausgeführt. *Befehlsfolge* steht für den Befehl, die Flags und die Parameter für den auszuführenden Prozess.

Die vollständige Syntax des Befehls **nice** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

Diese Task kann auch mit dem Befehl **smit nice** ausgeführt werden.

#### Priorität eines aktiven Prozesses ändern (Befehl renice):

Sie können einem aktiven Prozess, ausgehend von der Basisplanungspriorität, eine niedrigere oder höhere Priorität zuweisen. Führen Sie dazu in der Befehlszeile den Befehl **renice** aus. Mit diesem Befehl wird der *nice-Wert* eines Prozesses geändert.

**Anmerkung:** Zum Ausführen eines Prozesses mit einer höheren Priorität oder zum Ändern der Priorität eines nicht von Ihnen gestarteten Prozesses müssen Sie Rootberechtigung besitzen.

Geben Sie zum Ändern der Priorität für einen Prozess Folgendes ein:

```
renice Priorität -p Prozess-ID
```

*Priorität* steht hier für einen Wert zwischen -20 und 20. Je höher die Zahl ist, desto niedriger ist die Priorität. Wenn Sie die Priorität null verwenden, wird der Prozess mit der Basispriorität ausgeführt. Die *Prozess-ID* ist die PID, deren Priorität Sie ändern möchten.

Diese Task kann auch mit dem Befehl **smit renice** ausgeführt werden.

### Vordergrundprozesse abbrechen:

Wenn Sie einen Vordergrundprozess starten und sich danach entscheiden, den Prozess nicht zu Ende auszuführen, können Sie ihn durch Drücken der Unterbrechungstaste abbrechen. Hierbei handelt es sich gewöhnlich um die Tastenkombination Strg-C oder Strg-Rücktaste.

**Anmerkung:** Hintergrundprozesse können mit der Unterbrechungstaste (Strg-C) nicht abgebrochen werden. Zum Abbrechen eines Hintergrundprozesses müssen Sie den Befehl **kill** verwenden.

Die meisten einfachen Befehle werden so schnell ausgeführt, dass Sie gar keine Zeit haben, sie abzuberechen. Daher wird in den Beispielen im folgenden Abschnitt ein Befehl mit längerer Verarbeitungsdauer verwendet, nämlich der Befehl **find / -type f**. Dieser Befehl zeigt die Pfadnamen aller Dateien auf Ihrem System an. Für die in diesem Abschnitt beschriebenen Prozeduren ist es nicht erforderlich, näher auf den Befehl **find** einzugehen. Er ist lediglich ein Beispiel für das Arbeiten mit Prozessen.

Im folgenden Beispiel wird mit dem Befehl **find** ein Prozess gestartet. Dieser Prozess kann nach einigen Sekunden wie folgt mit der Unterbrechungstaste abgebrochen werden:

```
$ find / -type f
/usr/sbin/acct/lastlogin
/usr/sbin/acct/prctmp
/usr/sbin/acct/prdaily
/usr/sbin/acct/runacct
/usr/sbin/acct/sdisk
/usr/sbin/acct/shutacct INTERRUPT (Strg-C)
$ _
```

Die Eingabeaufforderung wird wieder angezeigt. Sie können nun einen anderen Befehl eingeben.

### Zugehörige Tasks:

„Liste der Steuertastenbelegungen für Terminal (Befehl stty)“ auf Seite 330

Mit dem Befehl **stty** können Sie Ihre Terminaleinstellungen anzeigen. Achten Sie insbesondere auf die Tasten, die das Terminal als Steuertasten verwendet.

### Tastaturbefehl für das Stoppen eines Vordergrundprozesses:

Es ist möglich, einen Prozess zu stoppen, ohne dass seine Prozess-ID (PID) aus der Prozesstabelle entfernt wird. Ein Vordergrundprozess kann durch Drücken der Tastenkombination Strg+Z gestoppt werden.

**Anmerkung:** Die Tastenkombination Strg-Z funktioniert in der Korn-Shell (**ksh**) und in der C-Shell (**cs**h), aber nicht in der Bourne-Shell (**bs**h).

### Gestoppten Prozess erneut starten:

Im Folgenden wird beschrieben, wie Sie einen Prozess, der mit der Tastenkombination Strg-Z gestoppt wurde, erneut starten können.

**Anmerkung:** Die Tastenkombination Strg-Z funktioniert in der Korn-Shell (**ksh**) und in der C-Shell (**cs**h), aber nicht in der Bourne-Shell (**bs**h). Zum Neustart eines gestoppten Prozesses müssen Sie entweder der Benutzer sein, der den Prozess gestartet hat, oder Rootberechtigung besitzen.

1. Geben Sie Folgendes ein, um alle aktiven und gestoppten, aber nicht vom System gelöschten Prozesse anzuzeigen:

```
ps -ef
```

Dieser Befehl kann mit dem Befehl **grep** verkettet werden, um die Liste der Prozesse, unter denen sich der erneut zu startende Prozess wahrscheinlich befindet, einzugrenzen. Geben Sie beispielsweise Folgendes ein, um eine **vi**-Sitzung erneut zu starten:

```
ps -ef | grep vi
```

Mit diesem Befehl werden ausschließlich die Zeilen der Ausgabe des Befehls **ps** angezeigt, die das Wort **vi** enthalten. Es wird eine ähnliche Ausgabe wie die folgende angezeigt:

UID	PID	PPID	C	STIME	TTY	TIME	COMMAND
root	1234	13682	0	00:59:53	-	0:01	vi test
root	14277	13682	1	01:00:34	-	0:00	grep vi

2. Suchen Sie in der Ausgabe des Befehls **ps** den erneut zu startenden Prozess, und notieren Sie die Prozess-ID. Die Prozess-ID im Beispiel ist 1234.
3. Geben Sie Folgendes ein, um dem gestoppten Prozess das Signal zur Fortsetzung zu senden:

```
kill -19 1234
```

Ersetzen Sie 1234 durch die PID Ihres Prozesses. -19 ist das Signal zur Fortsetzung des Prozesses. Mit diesem Befehl wird der Prozess im Hintergrund erneut gestartet. Wenn der Prozess im Hintergrund ausgeführt werden kann, ist die Prozedur hiermit beendet. Muss der Prozess im Vordergrund ausgeführt werden (wie z. B. eine **vi**-Sitzung), müssen Sie den nächsten Schritt ausführen.

4. Geben Sie Folgendes ein, um den Prozess im Vordergrund auszuführen:

```
fg 1234
```

Ersetzen Sie auch hier 1234 durch die PID Ihres Prozesses. Der Prozess wird jetzt im Vordergrund ausgeführt. (Die Sitzung des Editors **vi** ist aktiv.)

### Prozess für spätere Ausführung einplanen:

Ein Prozess kann für die Ausführung im Hintergrund zu einem geplanten Zeitpunkt als *Stapelprozess* definiert werden.

Mit den Befehlen **at** und **smit** können Sie die Namen der gewünschten Befehle und den Zeitpunkt für deren Ausführung angeben.

**Anmerkung:** Die Dateien `/var/adm/cron/at.allow` und `/var/adm/cron/at.deny` steuern, ob Sie den Befehl **at** verwenden können. Ein Benutzer mit Rootberechtigung kann diese Dateien erstellen, editieren und löschen. Die Einträge in diesen Dateien sind Benutzeranmeldungen mit einem Namen pro Zeile. Das folgende Beispiel zeigt eine Datei `at.allow`:

```
root
nick
dee
sarah
```

Wenn die Datei `at.allow` vorhanden ist, können nur die Benutzer, deren Anmeldenamen in dieser Datei aufgeführt sind, den Befehl **at** verwenden. Ein Systemadministrator kann einen Benutzer für die Verwendung des Befehls **at** explizit sperren, indem er den Anmeldenamen des Benutzers in die Datei `at.deny` aufnimmt. Wenn nur die Datei `at.deny` vorhanden ist, kann jeder Benutzer, dessen Name nicht in der Datei aufgeführt ist, den Befehl **at** verwenden.

Sie können den Befehl **at** nicht verwenden, wenn die folgenden Bedingungen zutreffen:

- Die Dateien `at.allow` und `at.deny` sind nicht vorhanden (nur `root`).
- Die Datei `at.allow` ist vorhanden, der Anmelde-name des Benutzers ist jedoch nicht eingetragen.

- Die Datei `at.deny` ist vorhanden, und der Anmeldenname des Benutzers ist eingetragen.

Wenn die Datei `at.allow` nicht vorhanden ist oder die Datei `at.deny` nicht vorhanden oder leer ist, können nur Benutzer mit Rootberechtigung mit dem Befehl `at` einen Job übergeben.

Die Syntax des Befehls `at` lässt Datums-, Zeit- und Tagesangaben sowie Inkrementzeichenfolgen für den gewünschten Startzeitpunkt des Prozesses zu. Außerdem können Sie angeben, welche Shell oder welche Warteschlange verwendet werden soll. Im Folgenden werden typische Verwendungsbeispiele für den Befehl beschrieben.

Wenn Ihr Anmeldenname beispielsweise `joyce` ist und Sie ein Script mit dem Namen `WorkReport` besitzen, das um 0:00 Uhr ausgeführt werden soll, gehen Sie wie folgt vor:

1. Geben Sie die gewünschte Startzeit für die Ausführung des Programms ein:  
`at midnight`
2. Geben Sie die Namen der auszuführenden Programme ein. Drücken Sie nach jedem eingegebenen Namen die Eingabetaste. Nach der Eingabe des letzten Namens drücken Sie die Tastenkombination Strg-D (Dateiende), um das Ende der Liste zu kennzeichnen.  
`WorkReport^D`

Nach dem Drücken der Tastenkombination Strg-D zeigt das System Informationen ähnlich den folgenden an:

```
job joyce.741502800.a at Fri Jul 6 00:00:00 CDT 2002.
```

Dem Programm `WorkReport` wurde die Jobnummer `joyce.741502800.a` zugeordnet. Das Programm wird am 6. Juli um 0:00 Uhr ausgeführt.

3. Wenn Sie beispielsweise nur die Programme auflisten möchten, deren Ausführung Sie auf einen späteren Zeitpunkt festgelegt haben, geben Sie Folgendes ein:  
`at -l`

Das System zeigt Informationen ähnlich den folgenden an:

```
joyce.741502800.a Fri Jul 6 00:00:00 CDT 2002
```

Die vollständige Syntax können Sie der Beschreibung des Befehls `at` in der Veröffentlichung "Commands Reference" entnehmen.

### Zugehörige Tasks:

„Alle geplanten Prozesse auflisten (Befehle `at` und `atq`)“

Verwenden Sie das Flag `-l` mit dem Befehl `at` oder dem Befehl `atq`, um alle geplanten Prozesse aufzulisten.

„Prozess aus dem Zeitplan entfernen“ auf Seite 152

Mit dem Befehl `at` und dem Flag `-r` können Sie einen geplanten Prozess entfernen.

### Alle geplanten Prozesse auflisten (Befehle `at` und `atq`):

Verwenden Sie das Flag `-l` mit dem Befehl `at` oder dem Befehl `atq`, um alle geplanten Prozesse aufzulisten.

Beide Befehle geben dasselbe aus. Der Befehl `atq` kann jedoch in derselben Zeit, in der der Befehl `at` ausgeführt wird und nur die Anzahl der Prozesse in der Warteschlange anzeigt, die Prozesse sortieren.

Alle geplanten Prozesse können auf die folgenden Arten aufgelistet werden:

- von der Befehlszeile aus mit dem Befehl `at`,
- mit dem Befehl `atq`.

### Befehl `at`

Geben Sie zum Auflisten der geplanten Prozesse Folgendes ein:

```
at -l
```

Dieser Befehl listet alle geplanten Prozesse in Ihrer Warteschlange auf. Wenn Sie als Root angemeldet sind, listet dieser Befehl alle geplanten Prozesse für alle Benutzer auf. Einzelheiten zur Befehlssyntax finden Sie in der Beschreibung des Befehls **at** in der Veröffentlichung "Commands Reference".

### Befehl atq

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **atq**:

- Geben Sie zum Auflisten aller geplanten Prozesse in der Warteschlange Folgendes ein:  
atq
- Wenn Sie ein Benutzer mit Rootberechtigung sind, können Sie mit dem folgenden Befehl die geplanten Prozesse in der Warteschlange eines bestimmten Benutzers auflisten:  
atq Benutzername
- Geben Sie Folgendes ein, um die Anzahl der geplanten Prozesse in der Warteschlange aufzulisten:  
atq -n

### Zugehörige Tasks:

„Prozess für spätere Ausführung einplanen“ auf Seite 150

Ein Prozess kann für die Ausführung im Hintergrund zu einem geplanten Zeitpunkt als *Stapelprozess* definiert werden.

„Prozess aus dem Zeitplan entfernen“

Mit dem Befehl **at** und dem Flag **-r** können Sie einen geplanten Prozess entfernen.

### Prozess aus dem Zeitplan entfernen:

Mit dem Befehl **at** und dem Flag **-r** können Sie einen geplanten Prozess entfernen.

Die folgenden Beispiele veranschaulichen die Verwendung der Befehle **at** und **atq**:

1. Wenn Sie einen geplanten Prozess entfernen möchten, müssen Sie die Nummer des Prozesses kennen. Die Prozessnummer kann mit dem Befehl **at -l** oder dem Befehl **atq** abgerufen werden.
2. Wenn Sie die Nummer des zu entfernenden Prozesses kennen, geben Sie Folgendes ein:  
at -r Jobnummer

Diese Task kann auch mit dem Befehl **smi t rmat** ausgeführt werden.

### Zugehörige Tasks:

„Alle geplanten Prozesse auflisten (Befehle at und atq)“ auf Seite 151

Verwenden Sie das Flag **-l** mit dem Befehl **at** oder dem Befehl **atq**, um alle geplanten Prozesse aufzulisten.

„Prozess für spätere Ausführung einplanen“ auf Seite 150

Ein Prozess kann für die Ausführung im Hintergrund zu einem geplanten Zeitpunkt als *Stapelprozess* definiert werden.

### Hintergrundprozess entfernen (Befehl kill):

Wenn Sie den Vordergrundprozess nicht mit der Unterbrechungstaste stoppen können oder wenn Sie einen bereits gestarteten Hintergrundprozess abbrechen möchten, verwenden Sie den Befehl **kill**.

Damit Sie einen Prozess mit dem Befehl **kill** abbrechen können, müssen Sie die Prozess-ID kennen. Das allgemeine Format für den Befehl **kill** ist folgendermaßen:

```
kill Prozess-ID
```



### Anmerkung:

- Sie müssen zum Entfernen eines Prozesses Rootberechtigung besitzen oder der Benutzer sein, der den Prozess gestartet hat. Das Standardsignal, das mit dem Befehl **kill** an einen Prozess abgesetzt wird, ist -15 (SIGTERM).
  - Zum Löschen eines Geisterprozesses (Zombie) muss der zugehörige übergeordnete Prozess gelöscht werden.
1. Die Prozess-ID des zu löschenden Prozesses kann mit dem Befehl **ps** ermittelt werden. Dieser Befehl kann mit dem Befehl **grep** verkettet werden, um ausschließlich den gewünschten Prozess aufzulisten. Geben Sie Folgendes ein, um beispielsweise die Prozess-ID einer Sitzung des Editors vi abzurufen:

```
ps -l | grep vi
```

2. Im nachfolgenden Beispiel wird der Befehl **find** zur Ausführung im Hintergrund abgesetzt. Anschließend wird entschieden, den Prozess abzubrechen. Führen Sie den Befehl **ps** aus, um die PID-Nummern aufzulisten.

```
$ find / -type f > dir.paths &
[1] 21593
$ ps
 PID TTY TIME COMMAND
 1627 pts3 0:00 ps
 5461 pts3 0:00 ksh
 17565 pts3 0:00 -ksh
 21593 pts3 0:00 find / -type f
$ kill 21593
$ ps
 PID TTY TIME COMMAND
 1627 pts3 0:00 ps
 5461 pts3 0:00 ksh
 17565 pts3 0:00 -ksh
[1] + Terminated 21593 find / -type f > dir.paths &
```

Der Befehl **kill 21593** stoppt den Hintergrundprozess **find**, und der zweite Befehl **ps** liefert keine Statusinformationen zu PID 21593. Das System gibt erst bei Eingabe des nächsten Befehls eine Abbruchnachricht aus, sofern es sich bei dem Befehl nicht um den Befehl **cd** handelt.

Mit dem Befehl **cd** können Sie Hintergrundprozesse abbrechen, wenn Sie beispielsweise einen Prozess versehentlich als Hintergrundprozess gestartet haben oder wenn die Ausführung des Prozesses zu lange dauert.

Die vollständige Syntax des Befehls **kill** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

Der Befehl **kill** kann auch wie folgt mit **smit** ausgeführt werden:

```
smit kill
```

## Befehlsübersicht für Befehle und Prozesse

Im Folgenden finden Sie eine Übersicht über die Befehle für Befehle und Prozesse.

Tabelle 55. Befehlsübersicht für Befehle

Eintrag	Beschreibung
<b>alias</b>	Shellbefehl, der eine Liste mit Aliasnamen in der Standardausgabe ausgibt.
<b>history</b>	Shellbefehl, der das Ereignisprotokoll anzeigt.
<b>man</b>	Zeigt Informationen zu Befehlen, Subroutinen und Dateien online an.
<b>whatis</b>	Beschreibt die Funktion, die ein Befehl ausführt.
<b>whereis</b>	Sucht die Quelldateien, Binärdateien und Dokumentation für installierte Programme.

Tabelle 56. Befehlsübersicht für Prozesse

Eintrag	Beschreibung
<b>at</b>	Führt Befehle zu einem späteren Zeitpunkt aus, listet alle geplanten Prozesse auf oder entfernt einen Prozess aus dem Zeitplan.
<b>atq</b>	Zeigt die Warteschlange mit den Jobs an, die ausgeführt werden sollen.
<b>kill</b>	Sendet ein Signal an aktive Prozesse.
<b>nice</b>	Führt einen Befehl mit einer niedrigeren oder höheren Priorität aus.
<b>ps</b>	Zeigt den aktuellen Status von Prozessen an.
<b>renice</b>	Ändert die Priorität aktiver Prozesse.

## Systemblockierungen verwalten

Die Verwaltung von Systemblockierungen ermöglicht Benutzern, unternehmenskritische Anwendungen unterbrechungsfrei auszuführen, und verbessert gleichzeitig die Verfügbarkeit von Anwendungen. Die Erkennung von Systemblockierungen warnt den Systemadministrator vor möglichen Problemen und erlaubt dem Administrator anschließend, sich als Root anzumelden oder einen Warmstart durchzuführen, um das Problem zu beheben.

### Befehl **shconf**

Der Befehl **shconf** wird aufgerufen, wenn die **Erkennung von Systemblockierungen** aktiviert ist. Der Befehl **shconf** konfiguriert, welche Ereignisse überwacht und welche Aktionen ausgeführt werden, wenn solche Ereignisse eintreten. Sie können jede der folgenden Aktionen, die zu prüfende Prioritätsstufe, das Zeitlimit, wenn keine Prozesse oder Threads mit niedrigerer oder gleicher Priorität ausgeführt werden, die Terminaleinheit für die Warnaktion und die Aktion für den Befehl **getty** angeben:

- Fehler in der Datei `errlog` protokollieren
- Warnung an der Systemkonsole (alphanumerische Konsole) oder einem nicht grafikfähigen Terminal (TTY) anzeigen
- Warmstart durchführen
- Speziellen **getty**-Prozess erzeugen, damit sich der Benutzer als Root anmelden und Befehle starten kann
- Befehl starten

Bei den Optionen **Befehl starten** und **Speziellen getty-Prozess erzeugen** startet die Erkennung von Systemblockierungen den Sonderbefehl **getty** bzw. den angegebenen Befehl mit der höchsten Priorität. Der Sonderbefehl **getty** gibt eine Warnung aus, in der mitgeteilt wird, dass es sich um einen **getty**-Prozess für Wiederherstellung handelt, der mit der Priorität 0 ausgeführt wird. In der folgenden Tabelle sind die verschiedenen Aktionen und die zugehörigen Standardparameter für die Erkennung kritischer Blockierungen zusammengefasst. Für jeden Typ von Erkennung ist nur eine einzige Aktion aktiviert.

Option	Aktivierung	Priorität	Zeitlimit (Sekunden)
Fehler in der Datei <code>errlog</code> protokollieren	inaktiviert	60	120
Warnung anzeigen	inaktiviert	60	120
<b>getty</b> -Prozess für Wiederherstellung erzeugen	aktiviert	60	120
Befehl starten	inaktiviert	60	120
Warmstart durchführen	inaktiviert	39	300

**Anmerkung:** Wenn die Option **getty-Prozess für Wiederherstellung an einer Konsole starten** aktiviert ist, fügt der Befehl **shconf** dem Befehl **getty** in der Datei `inittab`, die der Konsolanmeldung zugeordnet ist, das Flag **-u** hinzu.

Für die Erkennung von E/A-Verlusten können Sie das Zeitlimit festlegen und die folgenden Aktionen aktivieren:

Option	Aktivierung
Warnung anzeigen	inaktiviert
Warmstart durchführen	inaktiviert

## Dämon shdaemon

Der Dämon **shdaemon** ist ein Prozess, der vom Befehl **init** gestartet und mit der Priorität 0 (null) ausgeführt wird. Dieser Dämon ist für alles im Zusammenhang mit der Erkennung von Systemblockierungen zuständig. Er ruft Konfigurationsinformationen ab, baut Arbeitsstrukturen auf und startet die Erkennungsintervalle, die vom Benutzer festgelegt wurden.

### Zugehörige Konzepte:

„Erkennung von prioritätsbasierten Blockierungen“

AIX kann Blockierungen im System erkennen und versuchen, auf der Basis benutzerdefinierter Aktionen solche Situationen zu beheben.

„Erkennung von Blockierungen durch E/A-Verlust“ auf Seite 156

AIX kann Blockierungen im System erkennen und versuchen, auf der Basis benutzerdefinierter Aktionen solche Situationen zu beheben.

## Erkennung von Systemblockierungen konfigurieren

Sie können die Konfiguration für die Erkennung von Systemblockierungen mit dem Verwaltungstool SMIT verwalten.

Mit den Optionen in den SMIT-Menüs können Sie den Erkennungsmechanismus aktivieren und inaktivieren, den aktuellen Status der Funktion anzeigen und die aktuelle Konfiguration ändern oder anzeigen. Die Direktaufrufe für die Menüs für die Erkennung von Systemblockierungen sind im Folgenden beschrieben:

### **smit shd**

Erkennung von Systemblockierungen verwalten

### **smit shstatus**

Status der Erkennung von Systemblockierungen

### **smit shpriocfg**

Merkmale für die Erkennung von Problemen mit Prioritäten ändern/anzeigen

### **smit shreset**

Standardkonfiguration für Erkennung von Problemen mit Prioritäten wiederherstellen

### **smit shliocfg**

Merkmale für die Erkennung von E/A-Verlusten ändern/anzeigen

### **smit shlioreset**

Standardkonfiguration für die Erkennung von E/A-Verlusten wiederherstellen

Sie können die Erkennung von Systemblockierungen auch mit dem Befehl **shconf** verwalten.

## Erkennung von prioritätsbasierten Blockierungen

AIX kann Blockierungen im System erkennen und versuchen, auf der Basis benutzerdefinierter Aktionen solche Situationen zu beheben.

Alle Prozesse (oder Threads) werden mit einer Priorität ausgeführt. Diese Priorität ist eine Zahl aus dem Bereich 0-126 mit null als höchster Priorität und 126 als niedrigster Priorität. Die Standardpriorität für alle Threads ist 60. Die Priorität eines Prozesses kann von jedem Benutzer mit dem Befehl **nice** herabgesetzt werden. Benutzer mit Rootberechtigung können die Priorität eines Prozesses auch erhöhen.

Der Kernel-Scheduler weist immer dem ausführbaren Thread mit der höchsten Priorität eine CPU zu. Wenn genügend viele Threads mit einer hohen Priorität vorhanden sind, ist es deshalb möglich, dass diese die Maschine vollständig beanspruchen, so dass keine Threads mit einer niedrigeren Priorität ausgeführt werden können. Wenn die aktiven Threads eine höhere Priorität haben als 60 (die Standardpriorität), können alle normalen Shells und Anmeldungen so weit blockiert werden, dass das gesamte System blockiert zu sein scheint.

Die Funktion für die Erkennung von Systemblockierungen stellt einen Mechanismus bereit, mit dem solche Situationen erkannt werden können und mit dem der Systemadministrator das System wiederherstellen kann. Diese Funktion wird als Dämon (**shdaemon**) implementiert, der mit der höchsten Prozesspriorität ausgeführt wird. Dieser Dämon fragt den Kernel nach dem Thread mit der niedrigsten Priorität ab, der in einem bestimmten Zeitraum ausgeführt wurde. Wenn die Priorität über einem konfigurierten Schwellenwert liegt, kann der Dämon eine von mehreren Aktionen ausführen. Jede dieser Aktionen kann unabhängig aktiviert und so konfiguriert werden, dass sie bei einer bestimmten Priorität und in einem bestimmten Zeitintervall ausgelöst wird. Im Folgenden sind die Aktionen und ihre Standardeinstellungen aufgeführt:

Aktion	Standardm. aktiviert	Standard-priorität	Standard-zeitlimit	Standard-einheit
1) Fehler protokollieren	nein	60	2	
2) Konsolnachricht	nein	60	2	/dev/console
3) Anmeldeshell mit hoher Priorität	ja	60	2	/dev/tty0
4) Befehl mit hoher Priorität ausführen	nein	60	2	
5) Absturz und Warmstart	nein	39	5	

#### Zugehörige Konzepte:

„Systemblockierungen verwalten“ auf Seite 154

Die Verwaltung von Systemblockierungen ermöglicht Benutzern, unternehmenskritische Anwendungen unterbrechungsfrei auszuführen, und verbessert gleichzeitig die Verfügbarkeit von Anwendungen. Die Erkennung von Systemblockierungen warnt den Systemadministrator vor möglichen Problemen und erlaubt dem Administrator anschließend, sich als Root anzumelden oder einen Warmstart durchzuführen, um das Problem zu beheben.

### Erkennung von Blockierungen durch E/A-Verlust

AIX kann Blockierungen im System erkennen und versuchen, auf der Basis benutzerdefinierter Aktionen solche Situationen zu beheben.

Aufgrund von E/A-Fehlern kann der E/A-Pfad blockiert und damit weitere E/As in diesem Pfad beeinträchtigt werden. In einer solchen Situation ist es wichtig, dass das Betriebssystem den Benutzer warnt und benutzerdefinierte Aktionen ausführt. Im Rahmen der Erkennung von E/A-Verlusten und entsprechender Benachrichtigung überwacht der **shdaemon** mit der Hilfe des Logical Volume Manager die E/A-Puffer über einen gewissen Zeitraum und prüft, ob E/As schon zu lange anstehen. Wenn die Wartezeit den Schwellenwert überschreitet, der in der Datei **shconf** definiert ist, wird ein E/A-Verlust erkannt, und es werden weitere Aktionen eingeleitet. Die Informationen über den E/A-Verlust werden im Fehlerprotokoll dokumentiert. Basierend auf den Einstellungen in der Datei **shconf** muss unter Umständen auch ein Warmstart durchgeführt werden, um die Situation mit dem E/A-Verlust zu bereinigen.

Für die Erkennung von E/A-Verlusten können Sie das Zeitlimit festlegen und die folgenden Aktionen aktivieren:

Aktion	Standardmäßig aktiviert	Standardeinheit
Konsolnachricht	Nein	/dev/console
Absturz und Warmstart	Nein	-

Weitere Informationen zur Erkennung von Systemblockierungen finden Sie im Abschnitt „Systemblockierungen verwalten“ auf Seite 154.

### Zugehörige Konzepte:

„Systemblockierungen verwalten“ auf Seite 154

Die Verwaltung von Systemblockierungen ermöglicht Benutzern, unternehmenskritische Anwendungen unterbrechungsfrei auszuführen, und verbessert gleichzeitig die Verfügbarkeit von Anwendungen. Die Erkennung von Systemblockierungen warnt den Systemadministrator vor möglichen Problemen und erlaubt dem Administrator anschließend, sich als Root anzumelden oder einen Warmstart durchzuführen, um das Problem zu beheben.

## Prozessverwaltung

Der Prozess ist die Entität, die das Betriebssystem verwendet, um die Nutzung der Systemressourcen zu steuern. Mit *Threads* kann der Verbrauch der Prozessorzeit gesteuert werden, aber die meisten Systemverwaltungstools setzen trotzdem voraus, dass Sie auf den Prozess verweisen, in dem der Thread ausgeführt wird, und nicht auf den Thread selbst.

Es werden Tools für die folgenden Zwecke bereitgestellt:

- Erstellung, Abbruch, Identität und Ressourcennutzung von Prozessen beobachten.
  - Der Befehl **ps** wird verwendet, um Prozess-IDs, Benutzer, verbrauchte CPU-Zeit und andere Attribute anzuzeigen.
  - Der Befehl **who -u** meldet die Shellprozess-ID angemeldeter Benutzer zurück.
  - Der Befehl **svmon** wird verwendet, um den Realspeicherverbrauch von Prozessen anzuzeigen. (Informationen zum Befehl **svmon** finden Sie in der Veröffentlichung *Performance Toolbox Version 3: Guide and Reference*.)
  - Der Mechanismus des Befehls **acct** schreibt bei der Prozessbeendigung Datensätze, in denen die Ressourcennutzung des Prozesses zusammengefasst ist.
- Prioritätsstufe steuern, auf der ein Prozess mit anderen Prozessen um die CPU in Konkurrenz tritt.
  - Der Befehl **nice** bewirkt, dass ein Befehl mit einer bestimmten Prozesspriorität ausgeführt wird.
  - Der Befehl **renice** ändert die Priorität eines bestimmten Prozesses.
- Prozesse beenden, die nicht mehr steuerbar sind.
  - Der Befehl **kill** sendet ein Beendigungssignal an einen oder mehrere Prozesse.

### Zugehörige Konzepte:

„Systemabrechnung“ auf Seite 164

Mit dem Dienstprogramm für Systemabrechnung können Sie Informationen zur Verwendung verschiedener Systemressourcen für Einzelpersonen und Gruppen erfassen und zurückmelden.

## Prozessüberwachung

Als Systemadministrator können Sie Prozesse verwalten.

Der Befehl **ps** ist das primäre Tool für die Überwachung der Prozesse im System. Die meisten Flags des Befehls **ps** können zwei Kategorien zugeordnet werden:

- Flags, die angeben, welche Prozesstypen in der Ausgabe angezeigt werden sollen
- Flags, die angeben, welche Attribute dieser Prozesse angezeigt werden sollen

Die hilfreichsten Varianten des Befehls **ps** für die Systemverwaltung sind im Folgenden beschrieben:

**Eintrag****ps -ef****Beschreibung**

Listet alle Prozesse, die keine Kernelprozesse sind, mit der Benutzer-ID, der Prozess-ID, der aktuellen CPU-Belastung, der CPU-Gesamtbelastung und dem Befehl einschließlich Parametern auf, der den Prozess gestartet hat.

**ps -fu Benutzer-ID**

Listet alle Prozesse, deren Eigner *Benutzer-ID* ist, mit der Prozess-ID, der aktuellen CPU-Belastung, der CPU-Gesamtbelastung und dem Befehl einschließlich Parametern auf, der den Prozess gestartet hat.

Wenn Sie die Benutzer ermitteln möchten, die derzeit am meisten CPU-Zeit beanspruchen, geben Sie Folgendes ein:

```
ps -ef | egrep -v "STIME|$LOGNAME" | sort +3 -r | head -n 15
```

Dieser Befehl listet die 15 CPU-intensivsten Prozesse, die nicht Ihre eigenen sind, in absteigender Reihenfolge auf.

Die folgenden beiden Tabellen sind für speziellere Verwendungen bestimmt. Sie vereinfachen die Auswahl der **ps**-Flags, weil sie eine Zusammenfassung der Auswirkungen der einzelnen Flags enthalten.

## Flags für Prozessauswahl

	-A	-a	-d	-e	-G -g	-k	-p	-t	-U -u	a	g	t	x
Alle Prozesse	J	-	-	-	-	-	-	-	-	-	J	-	-
Keine Prozessgruppenleiter und keinem Terminal zugeordnet	-	J	-	-	-	-	-	-	-	-	-	-	-
Keine Prozessgruppenleiter	-	-	J	-	-	-	-	-	-	-	-	-	-
Keine Kernelprozesse	-	-	-	J	-	-	-	-	-	-	-	-	-
Mitglieder angegebener Prozessgruppen	-	-	-	-	J	-	-	-	-	-	-	-	-
Kernelprozesse	-	-	-	-	-	J	-	-	-	-	-	-	-
Prozesse, die in der Liste mit den Prozessnummern angegeben sind	-	-	-	-	-	-	J	-	-	-	-	-	-
Prozesse, die TTYs in der Liste zugeordnet sind	-	-	-	-	-	-	-	J ( <i>n</i> TTYs)	-	-	-	J (1 TTY)	-
Angegebene Benutzerprozesse	-	-	-	-	-	-	-	-	J	-	-	-	-
Prozesse mit Terminals	-	-	-	-	-	-	-	-	-	J	-	-	-
Prozesse, die keinem TTY zugeordnet sind	-	-	-	-	-	-	-	-	-	-	-	-	J

## Flags für Spaltenauswahl

Standard 1	-f	-l	-U -u	Standard 2	e	l	s	u	v	
PID	J	J	J	J	J	J	J	J	J	J
TTY	J	J	J	J	J	J	J	J	J	J
TIME	J	J	J	J	J	J	J	J	J	J
CMD	J	J	J	J	J	J	J	J	J	J
USER	-	J	-	-	-	-	-	-	J	-
UID	-	-	J	J	-	-	J	-	-	-
PPID	-	J	J	-	-	-	J	-	-	-
C	-	J	J	-	-	-	J	-	-	-
STIME	-	J	-	-	-	-	-	-	J	-
F	-	-	J	-	-	-	-	-	-	-
S/STAT	-	-	J	-	J	J	J	J	J	J
PIR	-	-	J	-	-	-	J	-	-	-
NI/NICE	-	-	J	-	-	-	J	-	-	-
ADDR	-	-	J	-	-	-	J	-	-	-
SIZE	-	-	-	-	-	-	-	-	J	-
SZ	-	J	-	-	-	J	-	J	-	-
WCHAN	-	-	J	-	-	-	J	-	-	-
RSS	-	-	-	-	-	-	J	-	J	J
SSIZ	-	-	-	-	-	-	-	J	-	-
%CPU	-	-	-	-	-	-	-	-	J	J
%MEM	-	-	-	-	-	-	-	-	J	J
PGIN	-	-	-	-	-	-	-	-	-	J
LIM	-	-	-	-	-	-	-	-	-	J
TSIZ	-	-	-	-	-	-	-	-	-	J
TRS	-	-	-	-	-	-	-	-	-	J
<i>Umgebung</i> (hinter dem Befehl)	-	-	-	-	-	J	-	-	-	-

Wenn der Befehl **ps** ohne Flags oder mit einem Flag für Prozessangabe, das mit einem Minuszeichen beginnt, angegeben wird, werden die Spalten angezeigt, die unter Standard 1 aufgelistet sind. Wird der Befehl mit einem Flag für Prozessangabe, das nicht mit einem Minuszeichen beginnt, angegeben, werden die unter Standard 2 aufgelisteten Spalten angezeigt. Die Flags **-u** und **-U** sind sowohl Flags für Prozessauswahl als auch Flags für Spaltenauswahl.

Im Folgenden wird der Inhalt der einzelnen Spalten kurz beschrieben:

Eintrag	Beschreibung
PID	Prozess-ID
TTY	Terminal oder Pseudoterminal, das dem Prozess zugeordnet ist
TIME	Kumulative verbrauchte CPU-Zeit in Minuten und Sekunden
CMD	Befehl, den der Prozess ausführt
USER	Anmeldename des Benutzers, zu dem der Prozess gehört
UID	Numerische Benutzer-ID des Benutzers, zu dem der Prozess gehört
PPID	ID des diesem Prozess übergeordneten Prozesses
C	Zuletzt verbrauchte CPU-Zeit
STIME	Uhrzeit, zu der der Prozess gestartet wurde, sofern dies in den letzten 24 Stunden stattgefunden hat. Andernfalls wird das Datum des Prozessesstarts angezeigt.
F	Achtstelliger Hexadezimalwert, der die Flags beschreibt, die dem Prozess zugeordnet sind (siehe detaillierte Beschreibung des Befehls <b>ps</b> )

Eintrag	Beschreibung
S/STAT	Status des Prozesses (siehe detaillierte Beschreibung des Befehls <b>ps</b> )
PRI	Aktueller Prioritätswert des Prozesses
NI/NICE	Prioritätszahl für den Prozess
ADDR	Segmentnummer des Prozessesstapels
SIZE	(Flag <b>-v</b> ) Virtuelle Größe des Datenabschnitts des Prozesses (in Kilobytes)
SZ	(Flags <b>-l</b> und <b>l</b> ) Größe des Kernimage des Prozesses in Kilobytes
WCHAN	Ereignis, auf das der Prozess wartet
RSS	Summe aus Anzahl Arbeitssegmentseiten und Anzahl Codesegmentseiten im Hauptspeicher mal 4
SSIZ	Größe des Kernel-Stacks
%CPU	Prozentualer Anteil des CPU-Verbrauchs seit Prozessesstart
%MEM	Nominal, der Prozentsatz des Realspeichers, der vom Prozess verbraucht wird. Dieser Messwert korreliert nicht mit anderen Speicherstatistiken.
PGIN	Anzahl der Seiteneinlagerungen, die durch Fehlseitenbedingungen verursacht werden. Da alle E/As als Fehlseitenbedingungen klassifiziert werden, ist dieser Wert im Prinzip ein Messwert für das E/A-Volumen.
LIM	Immer <b>xx</b>
TSIZ	Größe des Textabschnitts der ausführbaren Datei
TRS	Anzahl der Codesegmentseiten mal 4
Umgebung	Wert aller Umgebungsvariablen für den Prozess

## Prozesspriorität ändern

Wenn Sie einen Prozess ermittelt haben, der zu viel CPU-Zeit beansprucht, können Sie die effektive Priorität dieses Prozesses reduzieren, indem Sie seine Prioritätszahl mit dem Befehl **renice** erhöhen.

Beispiel:

```
renice +5 ProzID
```

Die Prioritätszahl von *ProzID* erhöht den normalen Prioritätswert 20 für einen Vordergrundprozess auf 25. Sie müssen Rootberechtigung besitzen, um die Prioritätszahl des Prozesses *ProzID* auf 20 zurückzusetzen. Geben Sie Folgendes ein:

```
renice -5 ProzID
```

## Prozessbeendigung

Normalerweise verwenden Sie den Befehl **kill**, um einen Prozess zu beenden.

Der Befehl **kill** sendet ein Signal an den jeweiligen Prozess. Je nach Typ des Signals und Charakter des Programms, das in dem Prozess ausgeführt wird, wird der Prozess entweder beendet, oder er bleibt aktiv. Die folgenden Signale werden gesendet:

Eintrag	Beschreibung
SIGTERM	(Signal 15) ist eine Beendigungsanforderung an das Programm. Wenn das Programm eine Signalaroutine für SIGTERM hat, die die Anwendung nicht wirklich beendet, hat dieser Befehl <b>kill</b> möglicherweise keine Wirkung. Dies ist das Standardsignal, das von <b>kill</b> gesendet wird.
SIGKILL	(Signal 9) ist eine Anweisung, den Prozess unverzüglich zu beenden. Dieses Signal kann nicht abgefangen oder ignoriert werden.

Gewöhnlich ist das Signal SIGTERM dem Signal SIGKILL vorzuziehen. Wenn das Programm eine Routine für SIGTERM hat, kann diese den Prozess ordnungsgemäß bereinigen und beenden. Geben Sie Folgendes ein:

```
kill -term Prozess-ID
```

(**-term** kann weggelassen werden.) Wenn der Prozess nicht auf das Signal SIGTERM reagiert, geben Sie Folgendes ein:

```
kill -kill Prozess-ID
```



Möglicherweise finden Sie in Ihrer Prozesstabelle gelegentlich nicht mehr aktive Prozesse, auch *Geisterprozesse* genannt. Diese Prozesse werden nicht mehr ausgeführt, haben keinen zugeordneten Systemspeicherbereich, behalten aber weiterhin ihre PID. Sie erkennen einen Geisterprozess in der Prozesstabelle daran, dass in der Spalte CMD für diesen Prozess der Wert <defunct> ausgewiesen wird. Beispiel:

```

UID PID PPID C STIME TTY TIME CMD
 .
 .
 .
lee 22392 20682 0 Jul 10 - 0:05 xclock
lee 22536 21188 0 Jul 10 pts/0 0:00 /bin/ksh
lee 22918 24334 0 Jul 10 pts/1 0:00 /bin/ksh
lee 23526 22536 22 0:00 <defunct>
lee 24334 20682 0 Jul 10 ? 0:00 aixterm
lee 24700 1 0 Jul 16 ? 0:00 aixterm
root 25394 26792 2 Jul 16 pts/2 0:00 ksh
lee 26070 24700 0 Jul 16 pts/3 0:00 /bin/ksh
lee 26792 20082 0 Jul 10 pts/2 0:00 /bin/ksh
root 27024 25394 2 17:10:44 pts/2 0:00 ps -ef

```

Geisterprozesse existieren so lange, bis der zugehörige Elternprozess beendet oder ein Systemabschluss durchgeführt und das System erneut gestartet wird. Im vorherigen Beispiel ist der übergeordnete Prozess (PPID) der Befehl **ksh**. Wenn die Korn-Shell beendet wird, wird der "gestorbene" Prozess aus der Prozesstabelle entfernt.

Manchmal sammelt sich eine Reihe solcher Prozesse in der Prozesstabelle an, wenn eine Anwendung mehrere untergeordnete Prozesse erzeugt hat und nicht beendet wurde. Sollte diese Situation zu einem Problem werden, ändern Sie einfach die Anwendung so, dass deren Subroutine **sigaction** das Signal **SIGCHLD** ignoriert.

#### Zugehörige Informationen:

sigaction command

### Prozess binden und Prozessbindung aufheben

Sie können einen Prozess an einen Prozessor binden und die Bindung eines Prozesses an einen Prozessor aufheben.

Wenn Sie einen Prozess an einen Prozessor binden möchten oder die Bindung eines Prozesses an einen Prozessor aufheben möchten, müssen Sie Rootberechtigung besitzen.

Auf Multiprozessorsystemen können Sie die folgenden Methoden verwenden, um einen Prozess an einen Prozessor zu binden oder die Bindung eines Prozesses an einen Prozessor aufzuheben:

- SMIT
- Befehlszeile

**Anmerkung:** Das Binden eines Prozesses an einen Prozessor kann zwar eine Leistungsverbesserung für den gebundenen Prozess bewirken (weil sich die Anzahl der Hardware-Cache-Fehler verringert), aber ein übermäßiger Gebrauch dieser Funktion kann auch dazu führen, dass einzelne Prozessoren überlastet werden, während andere Prozessoren nur selten genutzt werden. Die daraus resultierenden Engpässe könnten den Gesamtdurchsatz und die Gesamtleistung beeinträchtigen. Während des normalen Betriebs empfiehlt es sich, die Prozesse automatisch vom Betriebssystem zu Prozessoren zuordnen zu lassen, wodurch die Systemlast auf alle Prozessoren verteilt wird. Binden Sie nur solche Prozesse an Prozessoren, von denen Sie wissen, dass sie von der Ausführung in einem bestimmten Prozessor profitieren.

## Tasks für das Binden eines Prozesses bzw. das Aufheben einer Prozessbindung

Task	SMIT-Direktaufruf	Befehl oder Datei
Einen Prozess binden	<b>smit bindproc</b>	<b>bindprocessor -q</b>
Bindung eines Prozesses aufheben	<b>smit ubindproc</b>	<b>bindprocessor -u</b>

### Korrekturen für blockierte oder nicht erwünschte Prozesse:

Blockierte und nicht erwünschte Prozesse können Probleme bei Ihrem Terminal verursachen. Bei einigen Problemen werden Nachrichten am Bildschirm angezeigt, die Ihnen Informationen über die möglichen Ursachen liefern.

Für die Ausführung der folgenden Prozeduren benötigen Sie ein zweites Terminal, einen Modem oder eine Netzanmeldung. Wenn Ihnen keine dieser Möglichkeiten zur Verfügung steht, können Sie das Terminalproblem durch einen Warmstart Ihrer Maschine beheben.

Wählen Sie die geeignete Prozedur für die Behebung Ihres Terminalproblems:

*Von Prozessen verwendete Terminals freigeben:*

Sie können blockierte oder nicht erwünschte Prozesse stoppen.

Gehen Sie wie folgt vor, um blockierte oder nicht erwünschte Prozesse zu identifizieren und zu stoppen:

1. Geben Sie den folgenden Befehl **ps** ein, um die aktiven Prozesse am Bildschirm anzuzeigen:

```
ps -ef | pg
```

Der Befehl **ps** zeigt den Prozessstatus an. Das Flag **-e** schreibt Informationen zu allen Prozessen (mit Ausnahme von Kernelprozessen), und das Flag **f** generiert eine vollständige Liste der Prozesse mit dem Befehlsnamen und den Parametern, die beim Erstellen des Prozesses verwendet wurden. Der Befehl **pg** begrenzt die Ausgabe auf jeweils eine Seite, so dass die Informationen nicht aus dem Sichtbereich verschoben werden.

Zu den verdächtigen Prozessen gehören System- und Benutzerprozesse, die übermäßig viele Systemressourcen, z. B. CPU oder Plattenspeicherplatz, beanspruchen. Systemprozesse wie **sendmail**, **routed** und **lpd** sind häufig solche "Ausreißer". Verwenden Sie den Befehl **ps -u**, um die CPU-Belastung zu überprüfen.

2. Stellen Sie mit dem Befehl **who** fest, wer Prozesse auf dieser Maschine ausführt:

```
who
```

Der Befehl **who** zeigt Informationen zu allen derzeit am System angemeldeten Benutzern an, z. B. den Anmeldenamen, den Namen der Workstation, das Datum und die Uhrzeit der Anmeldung.

3. Stellen Sie fest, ob ein Benutzerprozess gestoppt oder ausgesetzt oder die Priorität eines Benutzerprozesses geändert werden muss.

**Anmerkung:** Zum Stoppen von Prozessen, die nicht Ihre eigenen sind, müssen Sie Rootberechtigung besitzen. Wenn Sie einen Benutzerprozess beenden oder die Priorität eines Benutzerprozesses ändern, wenden Sie sich an den Prozesseigner, und erläutern Sie ihm, was Sie getan haben.

- Stoppen Sie den Prozess mit dem Befehl **kill**. Beispiel:

```
kill 1883
```

Der Befehl **kill** sendet ein Signal an einen aktiven Prozess. Zum Stoppen des Prozesses geben Sie die Prozess-ID (PID) an, die in diesem Beispiel 1883 lautet. Verwenden Sie den Befehl **ps**, um die PID-Nummern von Befehlen zu ermitteln.

- Setzen Sie den Prozess aus, und führen Sie ihn durch Angabe eines Et-Zeichens (&) im Hintergrund aus. Beispiel:

```
/u/bin1/prog1 &
```

Das Et-Zeichen (&) signalisiert, dass der Prozess im Hintergrund ausgeführt werden soll. In einem Hintergrundprozess wartet die Shell nicht auf die Beendigung des Befehls, bevor sie die Shellvorgabeaufforderung wieder anzeigt. Wenn die Ausführung eines Prozesses mehr als ein paar Sekunden dauert, führen Sie den Befehl im Hintergrund aus, indem Sie am Ende der Befehlszeile ein Et-Zeichen (&) eingeben. Im Hintergrund ausgeführte Jobs werden mit dem normalen Befehl **ps** angezeigt.

- Ändern Sie mit dem folgenden Befehl **renice** die Priorität von Prozessen, die zu viele Systemressourcen beanspruchen:

```
renice 20 1883
```

Der Befehl **renice** ändert die Planungspriorität eines oder mehrerer aktiver Prozesse. Je höher die Nummer, desto niedriger ist die Priorität. 20 ist die niedrigste Priorität.

Der vorherige Beispielbefehl **renice** teilt dem Prozess mit der Nummer 1883 die niedrigste Priorität zu. Der Prozess wird ausgeführt, sobald ein wenig Prozessorzeit verfügbar ist.

*Auf angezeigte Nachrichten antworten:*

Verwenden Sie diese Prozedur, um auf Nachrichten, die am Bildschirm angezeigt werden, zu reagieren und eine Wiederherstellung durchzuführen.

1. Stellen Sie sicher, dass die Umgebungsvariable **DISPLAY** korrekt definiert ist. Verwenden Sie eine der folgenden Methoden, um die Umgebungsvariable **DISPLAY** zu prüfen:

- Verwenden Sie den Befehl **setenv**, um die Umgebungsvariablen anzuzeigen.

```
setenv
```

Der Befehl **setenv** zeigt die geschützte Statusumgebung an, die bei Ihrer Anmeldung eingerichtet wurde.

Stellen Sie fest, ob die Variable **DISPLAY** gesetzt wurde. Im folgenden Beispiel erscheint die Variable **DISPLAY** nicht, d. h., die Variable **DISPLAY** ist auf keinen bestimmten Wert gesetzt.

```
SYSENVIRON:
NAME=casey
TTY=/dev/pts/5
LOGNAME=casey
LOGIN=casey
```

#### **ODER**

- Ändern Sie den Wert der Variablen **DISPLAY**. Wenn Sie die Variable beispielsweise auf die Maschine mit dem Namen **bastet** und Terminal 0 setzen möchten, geben Sie Folgendes ein:

```
DISPLAY=bastet:0
export DISPLAY
```

Wenn die Umgebungsvariable **DISPLAY** nicht auf einen bestimmten Wert gesetzt ist, wird standardmäßig **unix:0** (die Konsole) verwendet. Der Wert der Variablen hat das Format **Name:Nummer**, wobei **Name** für den Hostnamen einer bestimmten Maschine steht und **Nummer** für die Nummer des X-Servers des angegebenen Systems.

2. Setzen Sie das Terminal mit dem folgenden Befehl **stty** auf seine Standardeinstellungen zurück:

```
stty sane
```

Der Befehl **stty sane** stellt die Funktionstüchtigkeit der Terminaltreiber wieder her. Der Befehl gibt einen entsprechenden Rücksetzcode für das Terminal aus der Datei **/etc/termcap** (oder aus **/usr/share/lib/terminfo**, sofern verfügbar) aus.

3. Wenn der Rücksetzcode nicht ordnungsgemäß funktioniert, setzen Sie das Terminal wie folgt zurück:

```
^J stty sane ^J
```

^J steht für die Tastenkombination Strg-J.

## Mehrere Warteschlangen über die Umgebungsvariablen RT\_MPC und RT\_GRQ ausführen:

Die Verwendung mehrerer Warteschlangen erhöht die Prozessoraffinität von Threads, aber es gibt Fälle, in denen Sie diesem Effekt entgegenwirken möchten.

Wenn es nur eine Ausführungswarteschlange gibt, ist ein Thread, der von einem anderen aktiven Thread aktiviert wird, normalerweise in der Lage, die CPU, in der der aktivierende Thread ausgeführt wird, sofort zu verwenden. Wenn mehrere Ausführungswarteschlangen verwendet werden, kann sich der aktivierte Thread in der Ausführungswarteschlange einer anderen CPU befinden, die den aktivierenden Thread erst dann benachrichtigen kann, wenn die nächste Planungsentscheidung getroffen wird. Dies kann zu einer Verzögerung von 10 ms führen.

Dieses Szenario gleicht den Szenarios in früheren Releases dieses Betriebssystems bei der Verwendung der Option `bindprocessor`. Wenn alle CPUs ständig ausgelastet sind und mehrere voneinander abhängige Threads aktiviert werden, gibt es zwei Optionen.

- Die erste Option, bei der eine Ausführungswarteschlange verwendet wird, ist, die Umgebungsvariable `RT_GRQ=ON` zu setzen, die bewirkt, dass nicht gebundene ausgewählte Threads aus der globalen Ausführungswarteschlange verschoben werden.
- Alternativ können Benutzer die Option mit dem Echtzeitkernel (Befehl `bosdebug -R on` und anschließend `bosboot` eingeben) und die Umgebungsvariable `RT_MPC=ON` für ausgewählte Prozesse verwenden. Es ist wichtig, ein Leistungsprotokoll Ihrer Systeme zu verwalten, um die Auswirkungen aller versuchten Optimierungen näher überwachen zu können.

## Systemabrechnung

Mit dem Dienstprogramm für Systemabrechnung können Sie Informationen zur Verwendung verschiedener Systemressourcen für Einzelpersonen und Gruppen erfassen und zurückmelden.

Diese Abrechnungsinformationen können verwendet werden, um Benutzern Gebühren für die verwendeten Systemressourcen zu berechnen und ausgewählte Aspekte des Systembetriebs zu überwachen. Das Abrechnungssystem liefert die Summen für die Ressourcennutzung, die von Mitgliedern der Gruppe `adm` definiert werden, und berücksichtigt diese bei der Gebührenabrechnung, sofern der Befehl `chargefee` verwendet wird.

Außerdem stellt das Abrechnungssystem Daten bereit, um die Angemessenheit der aktuellen Ressourcenzuordnungen zu bewerten, Ressourcengrenzen und -kontingente festzulegen, künftige Anforderungen vorherzusagen und Verbrauchsmaterial für Drucker und andere Einheiten zu bestellen.

Die folgenden Informationen unterstützen Sie beim Implementieren des Abrechnungsdienstprogramms in Ihrem System.

### Zugehörige Konzepte:

„Prozessverwaltung“ auf Seite 157

Der Prozess ist die Entität, die das Betriebssystem verwendet, um die Nutzung der Systemressourcen zu steuern. Mit *Threads* kann der Verbrauch der Prozessorzeit gesteuert werden, aber die meisten Systemverwaltungstools setzen trotzdem voraus, dass Sie auf den Prozess verweisen, in dem der Thread ausgeführt wird, und nicht auf den Thread selbst.

„Workload Manager“ auf Seite 514

Mit Workload Manager (WLM) kann der Systemadministrator wirkungsvoller steuern, wie der VMM (Virtual Memory Manager) des Schedulers und das Platten-E/A-Subsystem Prozessen Ressourcen zuordnen.

„Klassenbezogene Abrechnung“ auf Seite 522

Mit dem AIX-Dienstprogramm für das Abrechnungssystem können Sie Informationen zur Verwendung verschiedener Systemressourcen nach Benutzer, Gruppe oder WLM-Klasse erfassen und Berichte erstellen.

### Zugehörige Tasks:

„Überläufe im Dateisystem /var beheben“ auf Seite 480

Überprüfen Sie Folgendes, wenn die Speicherkapazität des Dateisystems (/var) erschöpft ist.

### Zugehörige Informationen:

AIX Version 6.1 Advanced Accounting Subsystem

## Abrechnungsdatenberichte

Nach der Erfassung der verschiedenen Typen von Abrechnungsdaten werden die Datensätze verarbeitet und in Berichte konvertiert.

Abrechnungsbefehle konvertieren die Datensätze automatisch in Exponentialschreibweise, wenn die Zahlen sehr hoch werden. Eine Zahl wird in dem folgenden Format in Exponentialschreibweise dargestellt:

*Basis+Exp*

*Basis-Exp*

Dies entspricht der *Basis*, multipliziert mit der *+Exp*- oder *-Exp*-Potenz von 10. Die Exponentialschreibweise 1,345e+9 entspricht beispielsweise  $1,345 \times 10^9$  bzw. 1.345.000.000. Die Exponentialschreibweise 1,345e-9 entspricht  $1,345 \times 10^{-9}$  bzw. 0,000000001345.

### Zugehörige Konzepte:

„Prozessabrechnungsdaten“ auf Seite 185

Das Abrechnungssystem erfasst Daten zur Ressourcennutzung für jeden Prozess während der Ausführung.

### Tagesabrechnungsberichte:

Wenn Sie einen Tagesbericht generieren möchten, verwenden Sie den Befehl **runacct**.

Dieser Befehl fasst Daten in einer ASCII-Datei mit dem Namen `/var/adm/acct/sum(x)/rprtMMTT` zusammen. *MMTT* steht für Monat und Tag, an dem der Bericht ausgeführt wird. Der Bericht umfasst Folgendes:

- Täglicher Bericht
- Täglicher Verwendungsbericht
- Tägliche Befehlszusammenfassung
- Monatliche Gesamtzusammenfassung für Befehle
- Letzte Anmeldung

### Täglicher Bericht:

Tägliche Abrechnungsberichte enthalten Daten über Verbindungsdauer, Prozesse, Plattenbelegung, Druckerauslastung und zu berechnende Gebühren.

Der Befehl **acctmerg** fügt unaufbereitete Abrechnungsdaten zu Verbindungsdauer, Prozessen, Plattenbelegung, Druckerauslastung und zu berechnenden Gebühren in tägliche Berichte ein. Aufgerufen vom Befehl **runacct** im Rahmen des täglichen Betriebs, erzeugt der Befehl **acctmerg** Folgendes:

**/var/adm/acct/nite(x)/dacct**

Ein temporärer Bericht, der erzeugt wird, wenn eine der Eingabedateien voll ist.

**/var/adm/acct/sum(x)/tacct**

Ein kumulativer Gesamtbericht im Format `tacct`. Diese Datei wird vom Befehl **monacct** verwendet, um die Monatsübersicht im ASCII-Format zu erzeugen.

Der Befehl **acctmrg** kann Datensätze vom ASCII-Format in das Binärformat und umgekehrt konvertieren und Datensätze aus unterschiedlichen Quellen zu einem einzigen Datensatz für jeden Benutzer zusammenführen. Weitere Informationen finden Sie in der Beschreibung des Befehls **acctmrg** in der Veröffentlichung "Commands Reference".

Die erste Zeile des täglichen Berichts beginnt mit den Anfangs- und Endzeiten für die im Bericht erfassten Daten, einer Liste von Ereignissen auf Systemebene, einschließlich vorhandener Systemabschlüsse, Warmstarts und Änderungen auf Ausführungsebene. Außerdem wird die Gesamtdauer aufgelistet, die die Gesamtanzahl der Minuten in der Abrechnungsperiode angibt (normalerweise 1440 Minuten, wenn der Bericht alle 24 Stunden ausgeführt wird). Der Bericht enthält die folgenden Informationen:

Eintrag	Beschreibung
LINE	Verwendete(s) Konsole, TTY oder PTY
MINUTES	Gesamtnutzungsdauer der Leitung in Minuten
PERCENT	Dauer der Leitungsnutzung während der Abrechnungsperiode in Prozent
# SESS	Anzahl der neuen gestarteten Anmeldesitzungen
# ON	Identisch mit # SESS
# OFF	Anzahl der Abmeldungen zuzüglich Leitungsunterbrechungen

### Täglicher Verwendungsbericht:

Der tägliche Verwendungsbericht ist ein zusammenfassender Bericht über die Systemnutzung pro Benutzer-ID in der Abrechnungsperiode.

Einige Felder sind in Haupt- und Nebenzeiten aufgeteilt. Dies wird vom Abrechnungsadministrator im Verzeichnis `/usr/lib/acct/holidays` definiert. Der Bericht enthält die folgenden Informationen:

Eintrag	Beschreibung
UID	Benutzer-ID
LOGIN NAME	Benutzername
CPU (PRIME/NPRIME)	Gesamt-CPU-Zeit für alle Prozesse des Benutzers in Minuten
KCORE (PRIME/NPRIME)	Von aktiven Prozessen verbrauchter Gesamtspeicher in Kilobytes mal Minute
CONNECT (PRIME/NPRIME)	Gesamtverbindungsdauer (wie lange der Benutzer angemeldet war) in Minuten
DISK BLOCKS	Durchschnittlich vom Benutzer verbrauchter Plattenspeicherplatz in allen Dateisystemen, für die die Abrechnung aktiviert ist
FEES	Mit dem Befehl <b>chargefee</b> eingegebene Gesamtgebühren
# OF PROCS	Gesamtanzahl der Prozesse, die zu diesem Benutzer gehören
# OF SESS	Anzahl unterschiedlicher Anmeldesitzungen für diesen Benutzer
# DISK SAMPLES	Anzahl der in der Abrechnungsperiode vorgenommenen Plattenstichproben. Wenn DISK BLOCKS null ist, ist auch dieser Wert gleich null.

### Abrechnungsbericht 'Tägliche Befehlszusammenfassung':

Der Bericht "Tägliche Befehlszusammenfassung" zeigt alle Befehle an, die in der Abrechnungsperiode ausgeführt wurden. Jeder eindeutige Befehlsname steht in einer separaten Zeile.

Die Tabelle ist nach TOTAL KCOREMIN (siehe folgende Beschreibung) sortiert. Die erste Zeile enthält die Gesamtinformationen für alle Befehle. Die für jeden Befehl aufgelisteten Daten sind für alle Ausführungen des Befehls in der Abrechnungsperiode kumulativ. Die Spalten in dieser Tabelle enthalten die folgenden Informationen:

Eintrag	Beschreibung
COMMAND NAME	Ausgeführter Befehl
NUMBER CMDS	Anzahl der Befehlsausführungen
TOTAL KCOREMIN	Für die Ausführung des Befehls verbrauchter Gesamtspeicher in Kilobytes mal Minute
TOTAL CPU-MIN	Vom Befehl verbrauchte CPU-Zeit in Minuten
TOTAL REAL-MIN	Abgelaufene Gesamtzeit für den Befehl in Minuten
MEAN SIZE-K	Durchschnittlicher Speicherverbrauch durch den Befehl pro CPU-Minute
MEAN CPU-MIN	Durchschnittliche Anzahl von CPU-Minuten pro Befehlsausführung
HOG FACTOR	Messwert für die Inanspruchnahme der CPU durch den Befehl, während dieser aktiv ist. Dieser Messwert gibt das Verhältnis von <b>TOTAL CPU-MIN</b> zu <b>TOTAL REAL-MIN</b> an.
CHARS TRNSFD	Anzahl der vom Befehl mit Systemlese- und Systemschreiboperationen übertragenen Zeichen
BLOCKS READ	Anzahl der vom Befehl gelesenen und geschriebenen physischen Blöcke

### Abrechnungsbericht 'Monatliche Befehlszusammenfassung':

Die vom Befehl **monacct** erstellte monatliche Befehlszusammenfassung enthält Informationen zu allen Befehlen, die seit der Erstellung des letzten monatlichen Berichts ausgeführt wurden.

Die Felder und Informationen haben dieselbe Bedeutung wie in der täglichen Befehlszusammenfassung.

#### Letzte Anmeldung:

Der Bericht "Letzte Anmeldung" enthält zwei Felder für jede Benutzer-ID. Das erste Feld ist **JJ-MM-TT** und zeigt die letzte Anmeldung für den angegebenen Benutzer an. Das zweite Feld ist der Name des Benutzeraccounts.

Wird im Datumsfeld 00-00-00 ausgewiesen, wurde keine Anmeldung der Benutzer-ID protokolliert.

#### Zusammenfassung der Abrechnungsberichte:

Sie können einen Bericht generieren, der die unaufbereiteten Abrechnungsdaten zusammenfasst.

Verwenden Sie zum Zusammenfassen der unaufbereiteten Abrechnungsdaten den Befehl **sa**. Dieser Befehl liest die unaufbereiteten Abrechnungsdaten, die gewöhnlich in der Datei `/var/adm/pacct` erfasst werden, und die aktuellen Nutzungssummendaten in der Datei `/var/adm/savacct`, sofern vorhanden. Er kombiniert diese Informationen zu einem neuen Nutzungsergebnisbericht und löscht die unaufbereitete Daten-datei, um Platz für weitere Datenerfassungen zu machen.

#### Voraussetzungen

Der Befehl **sa** erfordert eine Eingabedatei mit unaufbereiteten Abrechnungsdaten, wie z. B. die Datei `pacct` (Prozessabrechnungsdatei). Zum Erfassen unaufbereiteter Abrechnungsdaten muss ein Abrechnungssystem konfiguriert und aktiv sein.

#### Vorgehensweise

Mit dem Befehl **sa** können Prozessabrechnungsdaten zusammengefasst, angezeigt und gespeichert werden. Bei der einfachsten Verwendung des Befehls wird eine Liste mit Statistiken über jeden Prozess angezeigt, der in der Datei `pacct` protokolliert wurde, die gerade gelesen wird. Geben Sie zum Erzeugen einer solchen Liste Folgendes ein:

```
/usr/sbin/sa
```

Geben Sie Folgendes ein, um die Abrechnungsdaten zusammenzufassen und in der Übersichtsdatei zusammenzuführen:

```
/usr/sbin/sa -s
```

Der Befehl **sa** bietet zahlreiche zusätzliche Flags an, mit denen Sie angeben können, wie die Abrechnungsdaten verarbeitet und angezeigt werden. Weitere Informationen finden Sie in der Beschreibung des Befehls **sa**.

### Zugehörige Tasks:

„Abrechnungssystem konfigurieren“ auf Seite 176  
Sie können ein Abrechnungssystem konfigurieren.

### Monatlicher Bericht:

Sie können einen monatlichen Abrechnungsbericht generieren.

Der vom Dämon **cron** aufgerufene Befehl **monacct** erzeugt Folgendes:

Eintrag	Beschreibung
<code>/var/adm/acct/fiscal</code>	Ein regelmäßiger Ergebnisbericht, der vom Befehl <b>monacct</b> aus dem Bericht <code>/var/adm/acct/sum/tacct</code> erzeugt wird. Der Befehl <b>monacct</b> kann so konfiguriert werden, dass er monatlich oder am Ende eines Abrechnungszeitraums ausgeführt wird.

### Berichte zur Verbindungsdauer:

Zu Abrechnungsdatensätzen gehören Datensätze zu Anmeldungen, Abmeldungen, Systemabschlüssen und letzten Sitzungen.

Der Befehl **runacct** ruft die beiden Befehle **acctcon1** und **acctcon2** auf, um die Datensätze zu Anmeldungen, Abmeldungen und Systemabschlüssen zu verarbeiten, die in der Datei `/var/adm/wtmp` erfasst werden. Der Befehl **acctcon1** konvertiert diese Datensätze in Sitzungsdatensätze und schreibt sie in die Datei `/var/adm/acct/nite(x)/lineuse`. Anschließend konvertiert der Befehl **acctcon2** die Sitzungsdatensätze in einen Gesamtabrechnungssatz, `/var/adm/logacct`, den der Befehl **acctmrg** zu den täglichen Berichten hinzufügt. Weitere Informationen finden Sie in den Beschreibungen der Befehle **runacct**, **acctcon1** und **acctcon2** in der Veröffentlichung "Commands Reference".

Wenn Sie den Befehl **acctcon1** in der Befehlszeile ausführen, müssen Sie das Flag **-l** angeben, um den Leitungsbenutzungsbericht, `/var/adm/acct/nite(x)/lineuse`, zu erzeugen. Wenn Sie einen Gesamtsitzungsbericht für die Abrechnungsperiode erzeugen möchten, `/var/adm/acct/nite(x)/reboots`, verwenden Sie den Befehl **acctcon1** mit dem Flag **-o**.

Der Befehl **lastlogin** erzeugt einen Bericht, der das letzte Datum, an dem sich jeder Benutzer angemeldet hat, enthält. Weitere Informationen finden Sie in der Beschreibung des Befehls **lastlogin** in der Veröffentlichung "Commands Reference".

### Zugehörige Konzepte:

„Abrechnungsdaten zur Verbindungsdauer“ auf Seite 184

Daten zur Verbindungsdauer werden von den Befehlen **init** und **login** erfasst.

„Abrechnungsdaten zur Plattenbelegung“ auf Seite 186

Viele Abrechnungsdaten werden während des Verbrauchs der Ressourcen erfasst. Der Befehl **dodisk**, der nach den Angaben des Dämons **cron** ausgeführt wird, schreibt in regelmäßigen Abständen Datensätze zur Plattenbelegung für jeden Benutzer in die Datei `/var/adm/acct/nite(x)/dacct`.

### Abrechnungsbericht zur Plattenbelegung:

Die in der Datei `/var/adm/acct/nite(x)/dacct` erfassten Datensätze zur Plattenbelegung werden vom Befehl **acctmrg** in die täglichen Abrechnungsberichte eingefügt.

Informationen finden Sie in der Beschreibung des Befehls **acctmrg** in der Veröffentlichung "Commands Reference".



### Abrechnungsbericht zur Druckerauslastung:

Der ASCII-Datensatz in der Datei /var/adm/qacct kann in einen Gesamtabrechnungssatz konvertiert werden, der vom Befehl **acctmrg** dem täglichen Bericht hinzugefügt wird.

Informationen finden Sie in der Beschreibung des Befehls **acctmrg** in der Veröffentlichung "Commands Reference".

#### Zugehörige Konzepte:

„Abrechnungsdaten zur Druckerauslastung“ auf Seite 186

Die Erfassung der Druckerauslastungsdaten ist eine Gemeinschaftsarbeit des Befehls **enq** und des Warteschlangendämons.

### Gebührenabrechnungsbericht:

Wenn Sie den Befehl **chargefee** verwendet haben, um Benutzern Gebühren für Services, wie z. B. Datei-wiederherstellungen, Consulting oder Arbeitsmaterialien, zu berechnen, wird ein Gesamtabrechnungssatz in die Datei /var/adm/fee geschrieben. Diese Datei wird vom Befehl **acctmrg** zu den täglichen Berichten hinzugefügt.

Weitere Informationen finden Sie in den Beschreibungen der Befehle **chargefee** und **acctmrg** in der Veröffentlichung "Commands Reference".

#### Zugehörige Konzepte:

„Gebührenabrechnungsdaten“ auf Seite 187

Sie können einen Gesamtabrechnungssatz im ASCII-Format in der Datei /var/adm/fee erzeugen.

### Finanzabrechnungsberichte:

Die Finanzabrechnungsberichte werden im Allgemeinen monatlich mit dem Befehl **monacct** generiert.

Der Bericht wird in /var/adm/acct/fiscal(x)/fiscrptMM gespeichert, wobei MM für den Monat steht, in dem der Befehl **monacct** ausgeführt wird. Dieser Bericht enthält ähnliche Informationen wie die Tagesberichte, zusammengefasst für den gesamten Monat.

### Aktivitätsberichte zum Abrechnungssystem:

Sie können einen Bericht generieren, der Aufschluss über die Aktivität des Abrechnungssystems gibt.

Verwenden Sie zum Generieren eines Berichts über die Systemaktivität den Befehl **prtacct**. Dieser Befehl liest die Informationen aus einer Gesamtabrechnungsdatei (Dateiformat tacct) und erzeugt eine formatierte Ausgabe. Die Gesamtabrechnungsdateien enthalten die Tagesberichte über die Verbindungsdauer, die Verarbeitungszeit, die Plattenbelegung und die Druckerauslastung.

### Voraussetzungen

Der Befehl **prtacct** erfordert eine Eingabedatei im Dateiformat tacct. Dies setzt voraus, dass ein Abrechnungssystem konfiguriert und aktiv ist bzw. dass Sie das Abrechnungssystem zuvor verwendet haben.

### Vorgehensweise

Generieren Sie einen Bericht über die Systemaktivität. Geben Sie dazu Folgendes ein:

```
prtacct -f Spezifikation -v Überschrift Datei
```

*Spezifikation* steht für eine Liste, in der die Feldnummern bzw. -bereiche, die vom Befehl **acctmrg** verwendet werden, jeweils durch ein Komma voneinander getrennt aufgeführt sind. Das optionale Flag **-v** erzeugt eine ausführliche Ausgabe, in der Gleitkommazahlen in einer Schreibweise mit höherer Genauig-

keit angezeigt werden. *Überschrift* steht für den Titel, der im Bericht erscheinen soll. Diese Angabe ist optional. *Datei* steht für den vollständigen Pfadnamen der Gesamtabrechnungsdatei, die als Eingabe verwendet wird. Sie können mehrere Dateien angeben.

### Zugehörige Tasks:

„Abrechnungssystem konfigurieren“ auf Seite 176  
Sie können ein Abrechnungssystem konfigurieren.

### Unterstützung von Benutzernamen mit mehr als acht Zeichen:

Für die Gewährleistung der Abwärtskompatibilität mit allen Scripts ist die Unterstützung für lange Benutzernamen in der Abrechnung standardmäßig nicht aktiviert. Stattdessen werden alle Benutzer-IDs nach den ersten acht Zeichen abgeschnitten.

Damit die Unterstützung für lange Benutzernamen aktiviert werden kann, haben die meisten Befehle das zusätzliche Flag **-X**, das den Befehlen ermöglicht, Benutzer-IDs mit mehr als acht Zeichen zu akzeptieren und auszugeben (im ASCII- und im Binärformat). Wenn die Unterstützung für lange Benutzernamen aktiviert ist, verarbeiten Befehle und Scripts Dateien in den Verzeichnissen `/var/adm/acct/sumx`, `/var/adm/acct/nitex` und `/var/adm/acct/fiscalx` und verwenden nicht die Verzeichnisse `/var/adm/acct/sum`, `/var/adm/acct/nite` und `/var/adm/acct/fiscal`.

## Abrechnungsbefehle

Die Abrechnungsbefehle funktionieren auf verschiedene Arten.

### Einige Befehle

- erfassen Daten oder erzeugen Berichte für einen bestimmten Typ von Abrechnung: Verbindungsdauer, Prozesszeit, Plattenbelegung, Druckerauslastung oder Befehlsbenutzung.
- rufen andere Befehle auf. Der Befehl **runacct**, der gewöhnlich automatisch vom Dämon **cron** ausgeführt wird, ruft beispielsweise viele Befehle auf, die Abrechnungsdaten erfassen und verarbeiten und Berichte vorbereiten. Wenn Sie mit automatischer Abrechnung arbeiten möchten, müssen Sie zuerst den Dämon **cron** so konfigurieren, dass er den Befehl **runacct** ausführt. Weitere Informationen zum Konfigurieren des Dämons **cron** für die Übergabe von Befehlen in regelmäßigen Intervallen finden Sie in der Beschreibung des Befehls **crontab**. Weitere Informationen finden Sie in den Beschreibungen der Befehle **runacct**, **cron** daemon und **crontab** in der Veröffentlichung "Commands Reference".
- führen Wartungsfunktionen aus und stellen die Integrität aktiver Datendateien sicher.
- ermöglichen Mitgliedern der Gruppe `adm`, durch Eingabe eines Befehls an der Tastatur Gelegenheitsaufgaben auszuführen, wie z. B. bestimmte Datensätze anzeigen.
- ermöglichen einem Benutzer, bestimmte Informationen anzuzeigen. Es gibt nur einen Benutzerbefehl, den Befehl **acctcom**, der Übersichten über die Prozessabrechnung anzeigt.

### Automatisch ausgeführte Befehle:

Es gibt mehrere Befehle für die automatische Erfassung von Abrechnungsdaten.

Verschiedene Befehle, die gewöhnlich vom Dämon **cron** ausgeführt werden, erfassen Abrechnungsdaten automatisch. Es handelt sich dabei um die folgenden Befehle:

#### **runacct**

Führt die tägliche Hauptabrechnungsprozedur aus. Der Befehl **runacct**, der normalerweise vom Dämon **cron** zu Nebenzeiten eingeleitet wird, ruft mehrere andere Abrechnungsbefehle auf, um die aktiven Datendateien zu verarbeiten und Übersichten über die Befehls- und Ressourcennutzung nach Benutzername sortiert zu erzeugen. Er ruft außerdem den Befehl **acctmrg** auf, um tägliche Ergebnisberichtsdateien zu erzeugen, und den Befehl **ckpacct**, um die Integrität der aktiven Datendateien zu gewährleisten.

### **ckpacct**

Verwaltet die Größe der Datei `pacct`. Es empfiehlt sich, mehrere kleinere Dateien `pacct` zu verwenden, wenn Sie die Prozedur **runacct** nach einem Fehler während der Verarbeitung dieser Datensätze erneut starten müssen. Der Befehl **ckpacct** überprüft die Größe der aktiven Datendatei `/var/adm/pacct` und ruft, sofern die Datei größer als 500 Blöcke ist, den Befehl **turnacct switch** auf, um die Prozessabrechnung vorübergehend zu inaktivieren. Die Daten werden in eine neue `pacct`-Datei, `/var/adm/pacct x`, übertragen. ( $x$  steht für eine ganze Zahl, die jeweils um eins erhöht wird, wenn eine neue `pacct`-Datei erstellt wird.). Wenn die Anzahl freier Plattenblöcke auf einen Wert kleiner als 500 zurückgeht, ruft der Befehl **ckpacct** den Befehl **turnacct off** auf, um die Prozessabrechnung zu inaktivieren.

### **dodisk**

Ruft den Befehl **accdisk** und den Befehl **diskusg** oder **acctdusg** auf, um Datensätze zur Plattenbelegung in die Datei `/var/adm/acct/nite/dacct` zu schreiben. Diese Daten werden später in die täglichen Berichte eingefügt.

### **dodisk**

Ruft den Befehl **accdisk** und den Befehl **diskusg** oder **acctdusg** auf, um Datensätze zur Plattenbelegung in die Datei `/var/adm/acct/nite/dacct` zu schreiben. Diese Daten werden später in die täglichen Berichte eingefügt.

### **monacct**

Erzeugt aus täglichen Berichten eine periodenbezogene Zusammenfassung.

**sa1** Erfasst und speichert binäre Daten in der Datei `/var/adm/sa/sa tt`, wobei  $tt$  für den Tag im Monat steht.

**sa2** Schreibt einen täglichen Bericht in die Datei `/var/adm/sa/satt`, wobei  $tt$  für den Tag im Monat steht. Der Befehl entfernt Berichte aus der Datei `/var/adm/sa/satt`, die älter als eine Woche sind.

Weitere Befehle, die automatisch von anderen Prozeduren als dem Dämon **cron** ausgeführt werden:

### **startup**

Wenn der Befehl **startup** zur Datei `/etc/rc` hinzugefügt wird, leitet er Startprozeduren für das Abrechnungssystem ein.

### **shutacct**

Protokolliert die Zeit, zu der die Abrechnung inaktiviert wurde, indem er den Befehl **acctwtmp** aufruft, um eine Zeile in die Datei `/var/adm/wtmp` zu schreiben. Anschließend ruft der Befehl den Befehl **turnacct off** auf, um die Prozessabrechnung zu inaktivieren.

### **Tastaturbefehle:**

Ein Mitglied der Gruppe `adm` kann die folgenden Befehle über die Tastatur eingeben.

**ac** Gibt Datensätze zur Verbindungsdauer aus. Dieser Befehl wird für die Kompatibilität mit BSD-Systemen (Berkeley Software Distribution) bereitgestellt.

### **acctcom**

Zeigt Prozessabrechnungsübersichten an. Dieser Befehl steht auch den Benutzern zur Verfügung.

### **acctcon1**

Zeigt Übersichten über die Verbindungsdauer an. Sie müssen das Flag **-l** oder das Flag **-o** verwenden.

**accton** Aktiviert und inaktiviert die Prozessabrechnung.

### **chargefee**

Berechnet dem Benutzer eine vordefinierte Gebühr für ausgeführte Arbeitseinheiten. Die Gebühren werden vom Befehl **acctmerg** dem täglichen Bericht hinzugefügt.

### **fwtmp**

Konvertiert Dateien vom Binärformat in das ASCII-Format und umgekehrt.

**last** Zeigt Informationen zu vorherigen Anmeldungen an. Dieser Befehl wird für die Kompatibilität mit BSD-Systemen bereitgestellt.

**lastcomm**

Zeigt Informationen zu den letzten Befehlen an, die ausgeführt wurden. Dieser Befehl wird für die Kompatibilität mit BSD-Systemen bereitgestellt.

**lastlogin**

Zeigt den Zeitpunkt der letzten Anmeldung für jeden Benutzer an.

**pac** Bereitet Abrechnungsdatensätze für Drucker und Plotter vor. Dieser Befehl wird für die Kompatibilität mit BSD-Systemen bereitgestellt.

**prctmp**

Zeigt einen Sitzungssatz an.

**prtacct**

Zeigt Gesamtabrechnungsdateien an.

**sa** Fasst unaufbereitete Abrechnungsdaten zusammen, damit größere Mengen von Abrechnungsdaten besser verwaltet werden können. Dieser Befehl wird für die Kompatibilität mit BSD-Systemen bereitgestellt.

**sadc** Meldet verschiedene Aktionen des lokalen Systems zurück, z. B. Puffernutzung, E/A-Aktivitäten von Platten und Bändern, Aktivitätszähler für nicht grafikfähige Terminals (TTY) und Dateizugriffszähler.

**sar** Schreibt den Inhalt ausgewählter kumulativer Aktivitätszähler im Betriebssystem in die Standardausgabe. Der Befehl **sar** meldet nur lokale Aktivitäten zurück.

**time** Gibt die Echtzeit, die Benutzerzeit und die Systemzeit für die Ausführung eines Befehls aus.

**timex** Meldet die abgelaufene Zeit, die Benutzerzeit und die Ausführungszeit in Sekunden zurück.

**Zugehörige Konzepte:**

„Systemdaten erfassen und zurückmelden“ auf Seite 184

Sie können das System so konfigurieren, dass es automatisch Daten erfasst und Berichte generiert.

## Abrechnungsdateien

Die beiden Hauptabrechnungsverzeichnisse sind das Verzeichnis `/usr/sbin/acct`, in dem alle für die Ausführung des Abrechnungssystems erforderlichen Programme in der Programmiersprache C und Shellprozeduren gespeichert sind, und das Verzeichnis `/var/adm`, das die Daten-, Berichts- und Übersichtsdateien enthält.

Die Abrechnungsdatendateien gehören zu Mitgliedern der Gruppe `adm`, und alle aktiven Datendateien (z. B. `wtmp` und `pacct`) befinden sich im Ausgangsverzeichnis von `adm` (`/var/adm`).

### Abrechnungsdatendateien:

Die folgenden Dateien sind im Verzeichnis `/var/adm` gespeichert.

Eintrag	Beschreibung
/var/adm/diskdiag	Diagnosenachricht während der Ausführung der Plattenabrechnungsprogramme
/var/adm/dtmp	Ausgabe des Befehls <b>acctdusg</b>
/var/adm/fee	Ausgabe des Befehls <b>chargefee</b> in tacct-Datensätzen im ASCII-Format
/var/adm/pacct	Aktive Prozessabrechnungsdatei
/var/adm/wtmp	Aktive Prozessabrechnungsdatei
/var/adm/Spacct .mmtt	Prozessabrechnungsdateien für <i>mmtt</i> während der Ausführung des Befehls <b>runacct</b>

### Abrechnungsbericht und Übersichtsdateien:

Für die Aktivierung des Abrechnungssystems sind einige Unterverzeichnisse erforderlich.

Berichts- und Übersichtsdateien werden in einem Unterverzeichnis /var/adm/acct gespeichert. Vor der Aktivierung des Abrechnungssystems müssen Sie die folgenden Unterverzeichnisse erstellen.

#### **/var/adm/acct/nite(x)**

Enthält Dateien, die der Befehl **runacct** täglich wiederverwendet.

#### **/var/adm/acct/sum(x)**

Enthält die kumulativen Übersichtsdateien, die der Befehl **runacct** täglich aktualisiert.

#### **/var/adm/acct/fiscal(x)**

Enthält die monatlichen Übersichtsdateien, die der Befehl **monacct** erstellt.

### Zugehörige Tasks:

„Abrechnungssystem konfigurieren“ auf Seite 176  
Sie können ein Abrechnungssystem konfigurieren.

### Befehl **runacct** für die Abrechnung starten:

Sie können den Befehl **runacct** starten.

### Voraussetzungen

1. Das Abrechnungssystem muss installiert sein.
2. Sie müssen Root- oder Administratorgruppenberechtigung haben.

### Notes:

1. Wenn Sie den Befehl **runacct** ohne Parameter aufrufen, wird angenommen, dass der Befehl am jeweiligen Tag zum ersten Mal ausgeführt wird. Deshalb müssen Sie den Parameter *mmtt* beim erneuten Starten des Programms **runacct** angeben, damit Monat und Tag korrekt sind. Wenn Sie keinen Status angeben, liest das Programm **runacct** die Datei /var/adm/acct/nite(x)/statefile, um den Einstiegspunkt für die Verarbeitung zu bestimmen. Geben Sie zum Überschreiben der Datei /var/adm/acct/nite(x)/statefile in der Befehlszeile den gewünschten Status an.
2. Wenn Sie die folgende Task ausführen, müssen Sie unter Umständen den vollständigen Pfadnamen /usr/sbin/acct/runacct an Stelle des einfachen Befehlsnamens **runacct** verwenden.

### Vorgehensweise

Geben Sie zum Starten des Befehls **runacct** Folgendes ein:

```
nohup runacct 2> \
/var/adm/acct/nite/accterr &
```

Dieser Eintrag bewirkt, dass der Befehl alle Signale des Typs **INTR** und **QUIT** während der Hintergrundverarbeitung ignoriert. Er leitet alle Standardfehlerausgaben in die Datei /var/adm/acct/nite/accterr um.

## Befehl **runacct** für die Abrechnung erneut starten:

Wenn der Befehl **runacct** scheitert, können Sie ihn erneut starten.

Die Voraussetzungen für diese Prozedur sind im Folgenden beschrieben:

- Das Abrechnungssystem muss installiert sein.
- Sie müssen Root- oder Administratorgruppenberechtigung haben.

**Anmerkung:** Im Folgenden sind die häufigsten Ursachen für das Scheitern des Befehls **runacct** aufgelistet:

- Das System fährt herunter.
- Der Speicherplatz im Dateisystem /usr wird knapp.
- Die Datei /var/adm/wtmp enthält Datensätze mit inkonsistenten Datumsmarken.

Wenn der Befehl **runacct** scheitert, gehen Sie wie folgt vor:

1. Suchen Sie in der Datei /var/adm/acct/nite(x)/active *mmtt* nach Fehlernachrichten.
2. Wenn die aktive Datei und Sperrdateien in acct/nite vorhanden sind, überprüfen Sie die Datei accterr, an die Fehlernachrichten umgeleitet werden, wenn der Dämon **cron** den Befehl **runacct** aufruft.
3. Führen Sie alle erforderlichen Aktionen aus, um Fehler zu verhindern.
4. Starten Sie den Befehl **runacct** erneut.
5. Geben Sie Folgendes ein, um den Befehl **runacct** für ein bestimmtes Datum erneut zu starten:

```
nohup runacct 0601 2>> \
/var/adm/acct/nite/accterr &
```

Dieser Befehl startet das Programm **runacct** für den 1. Juni (0601). Das Programm **runacct** liest die Datei /var/adm/acct/nite/statefile, um festzustellen, welcher Anfangsstatus verwendet werden soll. Alle Standardfehlerausgaben werden an die Datei /var/adm/acct/nite/accterr angefügt.

6. Geben Sie Folgendes ein, um das Programm **runacct** mit dem angegebenen Status, z. B. MERGE, erneut zu starten:

```
nohup runacct 0601 MERGE 2>> \
/var/adm/acct/nite/accterr &
```

## Dateien des Befehls **runacct**:

Der Befehl **runacct** erzeugt Berichts- und Übersichtsdateien.

Die folgenden Berichts- und Übersichtsdateien, die der Befehl **runacct** erzeugt, sind von besonderem Interesse:

Eintrag	Beschreibung
/var/adm/acct/nite(x)/lineuse	Enthält Nutzungsstatistiken für jede Terminalleitung im System. Dieser Bericht ist besonders hilfreich für die Erkennung fehlerhafter Leitungen. Wenn das Verhältnis zwischen der Anzahl von Abmeldungen und Anmeldungen höher als 3:1 ist, ist die Wahrscheinlichkeit relativ hoch, dass eine Leitung fehlerhaft ist.
/var/adm/acct/nite(x)/daytacct /var/adm/acct/sum(x)/tacct	Enthält die Gesamtabrechnungsdatei für den vorherigen Tag. Enthält die Summierung der Datei nite/daytacct aller Tage und kann für die Gebührenabrechnung verwendet werden. Der Befehl <b>monacct</b> beginnt für jeden Monat bzw. jeden Abrechnungszeitraum eine neue Datei.
/var/adm/acct/sum(x)/cms	Enthält die Summierung der Befehlsübersichten aller Tage. Der Befehl <b>monacct</b> liest diese Binärversion der Datei und löscht sie anschließend. Die ASCII-Version ist nite/cms.
/var/adm/acct/sum(x)/daycms	Enthält die tägliche Befehlsübersicht. Eine ASCII-Version wird in nite/daycms gespeichert.
/var/adm/acct/sum(x)/loginlog /var/adm/acct/sum(x)/rprt <i>mmtt</i>	Enthält einen Datensatz zur letzten Verwendung jeder Benutzer-ID. Diese Datei enthält eine Kopie des täglichen Berichts, der vom Befehl <b>runacct</b> gespeichert wird.

### Dateien im Verzeichnis /var/adm/acct/nite(x):

Die folgenden Dateien sind im Verzeichnis /var/adm/acct/nite(x) gespeichert.

Eintrag	Beschreibung
active	In dieser Datei protokolliert der Befehl <b>runacct</b> den Fortschritt und gibt Warnungen und Fehlermeldungen aus. Die Datei <i>active.mmtt</i> ist eine Kopie der Datei <i>active</i> , die vom Programm <b>runacct</b> nach dem Feststellen eines Fehlers erstellt wird.
cms	Eine Gesamtbefehlsübersicht im ASCII-Format, die vom Befehl <b>prdaily</b> verwendet wird.
ctacct.mmtt	Gesamtabrechnungssätze für Verbindungen.
ctmp	Datensätze zur Verbindung der Sitzung.
daycms	Tägliche Befehlsübersicht im ASCII-Format, die vom Befehl <b>prdaily</b> verwendet wird.
dayacct	Gesamtabrechnungssätze für einen Tag.
dacct	Gesamtabrechnungssätze für die Plattenbelegung, die vom Befehl <b>ddisk</b> erstellt werden.
accterr	Diagnosenachrichten, die während der Ausführung des Befehls <b>runacct</b> erzeugt werden.
lastdate	Letzter Tag, an dem <b>runacct</b> ausgeführt wurde, im Format <i>date +%m%d</i> .
lockl	Wird verwendet, um die serielle Verwendung des Befehls <b>runacct</b> zu steuern.
lineuse	Bericht über die TTY-Leitungsnutzung, der vom Befehl <b>prdaily</b> verwendet wird.
log	Diagnosenachrichten des Befehls <b>accton1</b> .
log.mmtt	Identisch mit <b>log</b> , nachdem der Befehl <b>runacct</b> einen Fehler festgestellt hat.
reboots	Enthält die Anfangs- und Enddatumsangaben aus der Datei <i>wtmp</i> und eine Liste mit Systemwiederanläufen.
statefile	In dieser Datei wird der aktuelle Status während der Ausführung des Befehls <b>runacct</b> protokolliert.
tmpwtmp	Datei <i>wtmp</i> , die vom Befehl <b>wtmpfix</b> korrigiert wird.
wtmperror	Enthält <b>wtmpfix</b> -Fehlermeldungen.
wtmperr.mmtt	Identisch mit <i>wtmperror</i> , nachdem der Befehl <b>runacct</b> einen Fehler festgestellt hat.
wtmp.mmtt	Enthält die Datei <i>wtmp</i> des vorherigen Tages. Diese Datei wird während der Bereinigung durch den Befehl <b>runacct</b> entfernt.

### Dateien im Verzeichnis /var/adm/acct/sum(x):

Die folgenden Dateien sind im Verzeichnis /var/adm/acct/sum(x) gespeichert.

Eintrag	Beschreibung
cms	Gesamtbefehlsübersichtsdatei im Binärformat für den aktuellen Abrechnungszeitraum.
cmsprev	Befehlsübersichtsdatei ohne die letzte Aktualisierung.
daycms	Befehlsübersichtsdatei im Binärformat für den vorherigen Tag.
lastlogin	Datei, die vom Befehl <b>lastlogin</b> erstellt wird.
pacct.mmtt	Verknüpfte Version aller Dateien <i>pacct</i> für <i>mmtt</i> . Diese Datei wird nach dem Systemstart vom Befehl <b>remove</b> entfernt. Weitere Informationen finden Sie in der Beschreibung des Befehls <b>remove</b> in der Veröffentlichung "Commands Reference".
rprt.mmtt	Gespeicherte Ausgabe des Befehls <b>prdaily</b> .
tacct	Kumulative Gesamtabrechnungsdatei für den aktuellen Abrechnungszeitraum.
tacctprev	Identisch mit <i>tacct</i> mit Ausnahme der letzten Aktualisierung.
tacct.mmtt	Gesamtabrechnungsdatei für <i>mmtt</i> .

### Dateien im Verzeichnis /var/adm/acct/fiscal(x):

Die folgenden Dateien sind im Verzeichnis /var/adm/acct/fiscal(x) gespeichert.

Eintrag	Beschreibung
cms?	Gesamtbefehlsübersichtsdatei im Binärformat für den mit ? angegebenen Abrechnungszeitraum.
fiscrpt?	Ein dem vom Befehl <b>prdaily</b> erzeugten Bericht gleichender Bericht im Binärformat für den Abrechnungszeitraum, der für ? angegeben wird.
tacct?	Gesamtabrechnungsdatei im Binärformat für den Abrechnungszeitraum, der für ? angegeben wird.

## Abrechnungsdateiformate:

In der folgenden Tabelle sind die Ausgaben und Formate von Abrechnungsdateien zusammengefasst.

Eintrag	Beschreibung
wtmp	Erzeugt die aktive Prozessabrechnungsdatei. Das Format der Datei wtmp wird in der Datei utmp.h definiert. Informationen zur Datei utmp.h finden Sie im Abschnitt utmp.h.
ctmp	Erzeugt Datensätze zur Verbindung der Sitzung. Das Format wird in der Datei ctmp.h beschrieben.
pacct*	Erzeugt aktive Prozessabrechnungssätze. Das Format der Ausgabe wird in der Datei /usr/include/sys/acct.h definiert.
Spacct*	Erzeugt Prozessabrechnungsdateien für <i>mmtt</i> während der Ausführung des Befehls <b>runacct</b> . Das Format dieser Dateien wird in der Datei sys/acct.h definiert.
daytacct	Erzeugt Gesamtabrechnungssätze für einen Tag. Das Format der Datei wird in der Datei tacct definiert.
sum/tacct	Erzeugt eine Binärdatei, die die Befehlsübersichten für jeden Tag zusammenfasst. Das Format dieser Datei wird in der Header-Datei /usr/include/sys/acct.h definiert.
ptacct	Erzeugt verknüpfte Versionen der Dateien pacct. Das Format dieser Dateien wird in der Datei tacct definiert.
ctacct	Erzeugt Gesamtabrechnungssätze für Verbindungen. Die Ausgabe dieser Datei wird in der Datei tacct definiert.
cms	Erzeugt eine Gesamtbefehlsübersicht für die Abrechnung im Binärformat, die vom Befehl <b>prdaily</b> verwendet wird. Die ASCII-Version ist nite/cms.
daycms	Erzeugt eine tägliche Befehlsübersicht im Binärformat, die vom Befehl <b>prdaily</b> verwendet wird. Die ASCII-Version ist nite/daycms.

## Systemabrechnung verwalten

Für die Systemabrechnung können Sie verschiedene Tasks ausführen. Zu diesen Tasks gehören die Konfiguration eines Abrechnungssystems, die Anzeige der CPU-Belastung und die Anzeige der Abrechnungsprozesse.

### Abrechnungssystem konfigurieren:

Sie können ein Abrechnungssystem konfigurieren.

Für diese Prozedur benötigen Sie Rootberechtigung.

Die folgenden Informationen sind eine Übersicht über die Schritte, die Sie ausführen müssen, um ein Abrechnungssystem zu konfigurieren. Genauere Informationen finden Sie in den Beschreibungen der Befehle und Dateien, die in diesen Schritten erwähnt werden.

1. Mit dem Befehl **nulladm** können Sie sicherstellen, dass jede Datei die richtige Zugriffsberechtigung besitzt: Lese- (r) und Schreibberechtigung (w) für den Dateieigner und die Gruppe und Leseberechtigung (r) für andere. Geben Sie Folgendes ein:

```
/usr/sbin/acct/nulladm wtmp pacct
```

Mit diesem Befehl erhalten Sie Zugriff auf die Dateien pacct und wtmp.

2. Aktualisieren Sie die Datei /etc/acct/holidays, und fügen Sie die Stunden, die Sie als Hauptzeit festlegen, und die arbeitsfreien Tage für das Jahr ein.

**Anmerkung:** Kommentarzeilen können überall in der Datei erscheinen, solange das erste Zeichen in der Zeile ein Stern (\*) ist.



- a. Zum Definieren der Hauptzeit füllen Sie die Felder in der ersten Datenzeile (die erste Zeile, die nicht auf Kommentar gesetzt ist) aus. Verwenden Sie das 24-Stunden-Format. Diese Zeile setzt sich aus drei vierstelligen Feldern in der folgenden Reihenfolge zusammen:

- 1) Aktuelles Jahr
- 2) Beginn der Hauptzeit (*hhmm*)
- 3) Ende der Hauptzeit (*hhmm*)

Führende Leerzeichen werden ignoriert. Sie können Mitternacht als 0000 oder 2400 eingeben.

Wenn Sie beispielsweise das Jahr 2000 und eine Hauptzeit angeben möchten, die um 8:00 Uhr morgens beginnt und um 5:00 Uhr nachmittags endet, geben Sie Folgendes ein:

```
2000 0800 1700
```

- b. Zum Definieren der arbeitsfreien Tage im Unternehmen für das Jahr füllen Sie die nächste Datenzeile aus. Jede Zeile enthält vier Felder in der folgenden Reihenfolge:

- 1) Tag im Jahr
- 2) Monat
- 3) Tag im Monat
- 4) Beschreibung des arbeitsfreien Tags

Das Feld für den Tag im Jahr enthält die Nummer des Tages, auf den der arbeitsfreie Tag fällt. Der Wert muss eine Zahl zwischen 1 und 365 (366 in Schaltjahren) sein. Der erste Februar ist beispielsweise der Tag 32. Die anderen drei Felder sind nur zur Information bestimmt und werden als Kommentare behandelt.

Es folgt ein zweizeiliges Beispiel:

```
1 Jan 1 New Year's Day
332 Nov 28 Thanksgiving
```

3. Aktivieren Sie die Prozessabrechnung, indem Sie die folgende Zeile zur Datei `/etc/rc` hinzufügen oder das eventuell vorhandene Kommentarsymbol (`#`) am Anfang der Zeile löschen:

```
/usr/bin/su - adm -c /usr/sbin/acct/startup
```

Die Prozedur **startup** protokolliert die Zeit, zu der die Abrechnung aktiviert wurde, und bereinigt die Abrechnungsdateien des vorherigen Tages.

4. Geben Sie alle Dateisysteme an, die bei der Plattenabrechnung berücksichtigt werden sollen. Fügen Sie dazu die folgende Zeile zur Zeilengruppe für das jeweilige Dateisystem in der Datei `/etc/filesystems` hinzu:

```
account = true
```

5. Geben Sie die Datendatei an, die für Druckerdaten verwendet werden soll. Fügen Sie dazu die folgende Zeile zur Zeilengruppe für Warteschlangen in der Datei `/etc/qconfig` hinzu:

```
acctfile = /var/adm/qacct
```

6. Erstellen Sie als Benutzer `adm` ein Verzeichnis `/var/adm/acct/nite`, ein Verzeichnis `/var/adm/acct/fiscal` und ein Verzeichnis `/var/adm/acct/sum`, um die Datensätze für die tägliche Berichtsperiode und die Finanzberichtsperiode zu erfassen:

```
su - adm
cd /var/adm/acct
mkdir nite fiscal sum
exit
```

Verwenden Sie für lange Benutzernamen stattdessen die folgenden Befehle:

```
su - adm
cd /var/adm/acct
mkdir nitex fiscalx sumx
exit
```

7. Konfigurieren Sie die Prozeduren für die tägliche Abrechnung so, dass sie automatisch ausgeführt werden. Öffnen Sie dazu die Datei `/var/spool/cron/crontabs/adm` in einem Editor, und fügen Sie die Befehle **dodisk**, **ckpacct** und **runacct** hinzu. Beispiel:

```
0 2 * * 4 /usr/sbin/acct/dodisk
5 * * * * /usr/sbin/acct/ckpacct
0 4 * * 1-6 /usr/sbin/acct/runacct
 2>/var/adm/acct/nite/accterr
```

Fügen Sie für lange Benutzernamen stattdessen die folgenden Zeilen hinzu:

```
0 2 * * 4 /usr/sbin/acct/dodisk -X
5 * * * * /usr/sbin/acct/ckpacct
0 4 * * 1-6 /usr/sbin/acct/runacct -X
 2>/var/adm/acct/nitex/accterr
```

Die erste Zeile startet die Plattenabrechnung jeden Donnerstag (4) um 2:00 Uhr morgens (0 2). Die zweite Zeile startet jeden Tag (\*) jeweils 5 Minuten nach jeder vollen Stunde (5 \*) eine Überprüfung der Integrität der aktiven Datendateien. Die dritte Zeile führt die meisten Abrechnungsprozeduren aus und verarbeitet die aktiven Datendateien jeweils Montag bis Samstag (1-6) um 4:00 Uhr morgens (0 4). Wenn diese Zeiten nicht zu den Stunden passen, zu denen Ihr System arbeitet, müssen Sie die Einträge anpassen.

**Anmerkung:** Zum Editieren der Datei `/var/spool/cron/crontabs/adm` benötigen Sie Rootberechtigung.

8. Konfigurieren Sie die monatliche Abrechnungszusammenfassung so, dass sie automatisch ausgeführt wird. Fügen Sie dazu den Befehl **monacct** zur Datei `/var/spool/cron/crontabs/adm` hinzu. Geben Sie beispielsweise Folgendes ein:

```
15 5 1 * * /usr/sbin/acct/monacct
```

Fügen Sie für lange Benutzernamen stattdessen die folgende Zeile hinzu:

```
15 5 1 * * /usr/sbin/acct/monacct -X
```

Planen Sie diese Prozedur so, dass genug Zeit bleibt, den Bericht fertigzustellen. Dieses Beispiel startet die Prozedur jeweils am ersten Tag des Monats um 5:15 Uhr morgens.

9. Geben Sie Folgendes ein, um die editierte Datei `cron` zu übergeben:

```
crontab /var/spool/cron/crontabs/adm
```

### Zugehörige Konzepte:

„Befehl für die automatische Bereinigung von Dateisystemen“ auf Seite 397

Mit dem Befehl **skulker** können Sie Dateisysteme bereinigen und die nicht erwünschten Dateien entfernen.

„Systemdaten erfassen und zurückmelden“ auf Seite 184

Sie können das System so konfigurieren, dass es automatisch Daten erfasst und Berichte generiert.

„Aktivitätsberichte zum Abrechnungssystem“ auf Seite 169

Sie können einen Bericht generieren, der Aufschluss über die Aktivität des Abrechnungssystems gibt.

„Zusammenfassung der Abrechnungsberichte“ auf Seite 167

Sie können einen Bericht generieren, der die unaufbereiteten Abrechnungsdaten zusammenfasst.

### Zugehörige Tasks:

„Zugriff der Benutzer auf bestimmte Verzeichnisse einschränken“ auf Seite 397

Sie können Plattenspeicherplatz freigeben und diesen unter Umständen frei halten, indem Sie den Zugriff auf Verzeichnisse einschränken und die Plattenbelegung überwachen.

„Überlauf eines benutzerdefinierten Dateisystems korrigieren“ auf Seite 475

Verwenden Sie diese Prozedur, um ein überlaufendes benutzerdefiniertes Dateisystem zu korrigieren.

„Prozesszeit aktiver Abrechnungsprozesse anzeigen“ auf Seite 180

Sie können die Prozesszeit für aktive Prozesse anzeigen.

„Prozesszeit beendeter Abrechnungsprozesse anzeigen“ auf Seite 180

Sie können die Prozesszeit für beendete Prozesse anzeigen.

„CPU-Belastung für jeden Abrechnungsprozess anzeigen“ auf Seite 181

Mit dem Befehl **acctprc1** können Sie formatierte Berichte über die CPU-Belastung pro Benutzer anzeigen.

„Abrechnung der CPU-Belastung für jeden Benutzer anzeigen“ auf Seite 181

Mit einer Kombination der Befehle **acctprc1** und **prtacct** können Sie einen formatierten Bericht über die

CPU-Belastung pro Benutzer anzeigen.

„Abrechnungsdatensätze zur Drucker- oder Plotterauslastung anzeigen“ auf Seite 182

Mit dem Befehl **pac** können Sie Abrechnungsdatensätze zur Drucker- oder Plotterauslastung anzeigen.

#### Zugehörige Verweise:

„Abrechnungsbericht und Übersichtsdateien“ auf Seite 173

Für die Aktivierung des Abrechnungssystems sind einige Unterverzeichnisse erforderlich.

#### Aktivität des Abrechnungssystems anzeigen:

Mit dem Befehl **sar** können Sie formatierte Informationen zur Systemaktivität anzeigen.

Wenn Sie Statistiken zur Systemaktivität anzeigen möchten, muss der Befehl **sadc** aktiv sein.

**Anmerkung:** Typischerweise wird für die Ausführung des Befehls **sadc** ein Eintrag für den Befehl **sa1** in die Rootdatei `crontab` eingefügt. Der Befehl **sa1** ist eine Shellprozedurvariante des Befehls **sadc**, die für die Ausführung mit dem Dämon **cron** bestimmt ist.

Geben Sie Folgendes ein, um Informationen zur Systemaktivität anzuzeigen:

```
sar 2 6
```

Die erste Zahl steht für die Anzahl der Sekunden zwischen den Stichprobenintervallen und die zweite Zahl für die Anzahl der Anzeigeintervalle. Die Ausgabe dieses Befehls gleicht dem folgenden Beispiel:

```
arthurd 2 3 000166021000 05/28/92
14:03:40 %usr %sys %wio %idle
14:03:42 4 9 0 88
14:03:43 1 10 0 89
14:03:44 1 11 0 88
14:03:45 1 11 0 88
14:03:46 3 9 0 88
14:03:47 2 10 0 88
Average 2 10 0 88
```

Der Befehl **sar** bietet außerdem eine Reihe von Flags für die Anzeige zahlreicher Systemstatistiken an. Mit dem Flag **-A** können Sie alle verfügbaren Statistiken anzeigen. Eine Liste der verfügbaren Statistiken und der Flags für die Anzeige dieser Statistiken finden Sie in der Beschreibung des Befehls **sar**.

**Anmerkung:** Wenn ein Tagesbericht über die Systemaktivität nach `/var/adm/sa/satt` geschrieben werden soll, müssen Sie einen entsprechenden Eintrag in die Rootdatei `crontab` für den Befehl **sa2** einfügen. Der Befehl **sa2** ist eine Shellprozedurvariante des Befehls **sar**, die für die Ausführung mit dem Dämon **cron** bestimmt ist.

#### Aktivität des Abrechnungssystems während der Ausführung eines Befehls anzeigen:

Sie können formatierte Informationen zur Systemaktivität anzeigen, während ein bestimmter Befehl ausgeführt wird.

Die Flags **-o** und **-p** des Befehls **timex** setzen voraus, dass die Systemabrechnung aktiviert wird.

Sie können die Befehle **time** und **timex** verwenden, um formatierte Informationen über die Systemaktivität anzuzeigen, während ein bestimmter Befehl ausgeführt wird.

Geben Sie Folgendes ein, um die abgelaufene Zeit, die Benutzerzeit und die Systemausführungszeit für einen bestimmten Befehl anzuzeigen:

```
time Befehlsname
```

ODER

```
timex Befehlsname
```

Geben Sie Folgendes ein, um die Gesamtsystemaktivität (alle Datenelemente, die vom Befehl **sar** berichtet wurden) während der Ausführung eines bestimmten Befehls anzuzeigen:

```
timex -s Befehlsname
```

Der Befehl **timex** hat zwei zusätzliche Flags. Das Flag **-o** berichtet die Gesamtanzahl der Blöcke, die vom Befehl und allen seinen unterordneten Befehlen gelesen oder geschrieben werden. Das Flag **-p** listet alle Prozessabrechnungssätze für einen Befehl und alle seine untergeordneten Befehle auf.

### Prozesszeit aktiver Abrechnungsprozesse anzeigen:

Sie können die Prozesszeit für aktive Prozesse anzeigen.

Der Befehl **acctcom** liest Eingaben im Gesamtabrechnungssatzformat (acct-Dateiformat). Dies setzt voraus, dass die Prozessabrechnung aktiviert ist bzw. dass Sie die Prozessabrechnung zuvor verwendet haben.

Der Befehl **ps** bietet eine Reihe von Flags an, mit denen die angezeigten Informationen angepasst werden können.

Geben Sie Folgendes ein, um eine vollständige Liste aller aktiven Prozesse mit Ausnahme der Kernelprozesse zu erzeugen:

```
ps -ef
```

Sie können auch eine Liste aller Prozesse anzeigen, die Terminals zugeordnet sind. Geben Sie dazu Folgendes ein:

```
ps -al
```

In beiden Fällen wird eine Reihe von Spalten für jeden Prozess angezeigt, darunter die aktuelle CPU-Zeit für den Prozess in Minuten und Sekunden.

### Zugehörige Tasks:

„Abrechnungssystem konfigurieren“ auf Seite 176  
Sie können ein Abrechnungssystem konfigurieren.

### Prozesszeit beendeter Abrechnungsprozesse anzeigen:

Sie können die Prozesszeit für beendete Prozesse anzeigen.

Der Befehl **acctcom** liest Eingaben im Gesamtabrechnungssatzformat (acct-Dateiformat). Dies setzt voraus, dass die Prozessabrechnung aktiviert ist bzw. dass Sie die Prozessabrechnung zuvor verwendet haben.

Die Prozessabrechnungsfunktionen werden mit dem Befehl **startup** aktiviert, der gewöhnlich während der Systeminitialisierung über einen Aufruf in der Datei `/etc/rc` gestartet wird. Wenn die Prozessabrechnungsfunktionen aktiv sind, wird für jeden beendeten Prozess ein Datensatz in die Datei `/var/adm/pacct` (eine Gesamtabrechnungssatzdatei) geschrieben, der die Anfangs- und Endzeit für den Prozess enthält. Mit dem Befehl **acctcom** können die Informationen zur Prozesszeit aus einer Datei `pacct` angezeigt werden. Dieser Befehl hat eine Reihe von Flags, mit denen flexibel festgelegt werden kann, welche Prozesse angezeigt werden.

Wenn Sie beispielsweise alle Prozesse anzeigen möchten, die für eine Mindestanzahl von CPU-Sekunden oder länger ausgeführt wurden, verwenden Sie das Flag **-O** wie folgt:

```
acctcom -O 2
```

Dieser Befehl zeigt Datensätze für jeden Prozess an, der mindestens zwei Sekunden lang ausgeführt wurde. Wenn Sie keine Eingabedatei angeben, liest der Befehl **acctcom** Eingaben aus den Verzeichnis `/var/adm/pacct`.

#### **Zugehörige Tasks:**

„Abrechnungssystem konfigurieren“ auf Seite 176  
Sie können ein Abrechnungssystem konfigurieren.

#### **CPU-Belastung für jeden Abrechnungsprozess anzeigen:**

Mit dem Befehl **acctprc1** können Sie formatierte Berichte über die CPU-Belastung pro Benutzer anzeigen.

Der Befehl **acctprc1** erfordert eine Eingabe im Gesamtabrechnungssatzformat (acct-Dateiformat). Dies setzt voraus, dass die Prozessabrechnung aktiviert ist bzw. dass Sie die Prozessabrechnung zuvor verwendet haben.

Geben Sie Folgendes ein, um einen formatierten Bericht über die CPU-Belastung pro Prozess anzuzeigen:

```
acctprc1 </var/adm/pacct
```

#### **Zugehörige Tasks:**

„Abrechnungssystem konfigurieren“ auf Seite 176  
Sie können ein Abrechnungssystem konfigurieren.

#### **Abrechnung der CPU-Belastung für jeden Benutzer anzeigen:**

Mit einer Kombination der Befehle **acctprc1** und **prtacct** können Sie einen formatierten Bericht über die CPU-Belastung pro Benutzer anzeigen.

Der Befehl `../com.ibm.aix.cmds1/acctprc1.htm` erfordert eine Eingabe im Gesamtabrechnungssatzformat (acct-Dateiformat). Dies setzt voraus, dass die Prozessabrechnung aktiviert ist bzw. dass Sie die Prozessabrechnung zuvor verwendet haben.

Führen Sie die folgenden Schritte aus, um die CPU-Belastung für jeden Benutzer anzuzeigen:

1. Erzeugen Sie mit dem folgenden Befehl eine Ausgabedatei zur CPU-Belastung pro Prozess:

```
acctprc1 </var/adm/pacct >out.file
```

Die Datei `/var/adm/pacct` ist die Standardausgabe für Prozessabrechnungssätze. Sie können stattdessen eine `pacct`-Archivdatei angeben.

2. Erzeugen Sie mit dem folgenden Befehl aus der Ausgabe des vorherigen Schrittes eine binäre Gesamtabrechnungssatzdatei:

```
acctprc2 <out.file >/var/adm/acct/nite/daytacct
```

**Anmerkung:** Die Datei `daytacct` wird vom Befehl **acctmerg** mit anderen Gesamtabrechnungssätzen zum täglichen Summensatz (`/var/adm/acct/sum(x)/tacct`) zusammengeführt.

3. Verwenden Sie den Befehl `../com.ibm.aix.cmds4/prtacct.htm`, um einen formatierten Bericht über die CPU-Belastung pro Benutzer anzuzeigen:

```
prtacct </var/adm/acct/nite/daytacct
```

#### **Zugehörige Tasks:**

„Abrechnungssystem konfigurieren“ auf Seite 176  
Sie können ein Abrechnungssystem konfigurieren.

#### **Verbindungsdauer für Abrechnung anzeigen:**

Mit dem Befehl **ac** können Sie die Verbindungsdauer aller Benutzer, die Verbindungsdauer einzelner Benutzer oder die Verbindungsdauer pro Anmeldung anzeigen.

Der Befehl **ac** extrahiert Anmeldeinformationen aus der Datei `/var/adm/wtmp`. Folglich muss diese Datei vorhanden sein. Falls diese Datei nicht erstellt wurde, wird die folgende Fehlermeldung zurückgegeben:

```
No /var/adm/wtmp
```

Wenn die Datei zu voll wird, werden weitere `wtmp`-Dateien erstellt. Sie können Informationen zur Verbindungsdauer aus diesen Dateien anzeigen, indem Sie sie mit dem Flag **-w** angeben. Weitere Informationen finden Sie in der Beschreibung des Befehls **ac** in der Veröffentlichung "Commands Reference".

Geben Sie Folgendes ein, wenn Sie die Gesamtverbindungsdauer für alle Benutzer anzeigen möchten:

```
/usr/sbin/acct/ac
```

Dieser Befehl zeigt eine einzige Dezimalzahl an, die der Gesamtverbindungsdauer in Minuten für alle Benutzer entspricht, die sich während der Lebensdauer der aktuellen Datei `wtmp` angemeldet haben.

Geben Sie Folgendes ein, um die Gesamtverbindungsdauer für einen oder mehrere bestimmte Benutzer anzuzeigen:

```
/usr/sbin/acct/ac Benutzer1 Benutzer2 ...
```

Dieser Befehl zeigt eine einzige Dezimalzahl an, die der Gesamtverbindungsdauer in Minuten für alle angegebenen Benutzer für alle Anmeldungen während der Lebensdauer der aktuellen Datei `wtmp` entspricht.

Geben Sie Folgendes ein, um die Verbindungsdauer pro Benutzer sowie die Gesamtverbindungsdauer anzuzeigen:

```
/usr/sbin/acct/ac -p Benutzer1 Benutzer2 ...
```

Dieser Befehl zeigt für jeden angegebenen Benutzer eine Dezimalzahl an, die der Gesamtverbindungsdauer (in Minuten) für diesen Benutzer während der Lebensdauer der aktuellen Datei `wtmp` entspricht. Außerdem zeigt der Befehl eine Dezimalzahl an, die die Gesamtverbindungsdauer für alle angegebenen Benutzer angibt. Wenn Sie mit dem Befehl keinen Benutzer angeben, enthält die Liste alle Benutzer, die sich während der Lebensdauer der Datei `wtmp` angemeldet haben.

### Plattenspeicherplatzbelegung für die Abrechnung:

Mit dem Befehl **acctmrg** können Sie Informationen zur Plattenspeicherplatzbelegung anzeigen.

Zum Anzeigen von Informationen zur Plattenspeicherplatzbelegung benötigt der Befehl **acctmrg** Eingaben aus einer Datei `dacct` (Plattenabrechnung). Die Erfassung der Datensätze zur Plattenbelegungsabrechnung erfolgt mit dem Befehl **dadisk**.

Geben Sie Folgendes ein, um Informationen zur Plattenspeicherplatzbelegung anzuzeigen:

```
acctmrg -a1 -2,13 -h </var/adm/acct/nite(x)/dacct
```

Dieser Befehl zeigt Datensätze zur Plattenabrechnung an, darunter die Anzahl der 1-KB-Blöcke, die jeder einzelne Benutzer verbraucht hat.

**Anmerkung:** Der Befehl **acctmrg** liest immer aus der Standardeingabe und kann bis zu neun zusätzliche Dateien lesen. Wenn Sie die Eingabe nicht über eine Pipe an den Befehl weiterleiten, müssen Sie die Eingabe von einer Datei umleiten. Die restlichen Dateien können ohne Umleitung angegeben werden.

### Abrechnungsdatensätze zur Drucker- oder Plotterauslastung anzeigen:

Mit dem Befehl **pac** können Sie Abrechnungsdatensätze zur Drucker- oder Plotterauslastung anzeigen.

- Zum Erfassen von Informationen zur Druckerauslastung muss ein Abrechnungssystem konfiguriert und aktiv sein. Richtlinien hierzu finden Sie im Abschnitt „Abrechnungssystem konfigurieren“ auf Seite 176.

- Für den Drucker oder Plotter, für den Sie Abrechnungsdatensätze erfassen möchten, muss eine Klausel `acctfile=` in der Druckerzeilengruppe der Datei `/etc/qconfig` enthalten sein. Die in der Klausel `acctfile=` angegebene Datei muss Root bzw. der Gruppe `printq` Lese- und Schreibberechtigungen erteilen.
- Wenn das Flag `-s` des Befehls `pac` angegeben wird, schreibt der Befehl den Namen der Übersichtsdatei um, indem er `_sum` an den Pfadnamen anfügt, der in der Klausel `acctfile=` in der Datei `/etc/qconfig` angegeben ist. Diese Datei muss vorhanden sein und Root bzw. der Gruppe `printq` Lese- und Schreibberechtigungen erteilen.

Geben Sie Folgendes ein, um Informationen zur Druckerauslastung für alle Benutzer eines bestimmten Druckers anzuzeigen:

```
/usr/sbin/pac -PDrucker
```

Wenn Sie keinen Drucker angeben, wird der Standarddrucker mit der Umgebungsvariablen `PRINTER` angegeben. Ist die Variable `PRINTER` nicht definiert, wird standardmäßig `lp0` verwendet.

Geben Sie Folgendes ein, um Informationen zur Druckerauslastung für bestimmte Benutzer eines bestimmten Druckers anzuzeigen:

```
/usr/sbin/pac -PDrucker Benutzer1 Benutzer2 ...
```

Der Befehl `pac` bietet weitere Flags an, mit denen Sie steuern können, welche Informationen angezeigt werden.

#### Zugehörige Tasks:

„Abrechnungssystem konfigurieren“ auf Seite 176  
Sie können ein Abrechnungssystem konfigurieren.

#### Datei `holidays` aktualisieren:

Nach dem letzten arbeitsfreien Tag in der Liste bzw. nach dem Jahreswechsel ist die Datei `holidays` nicht mehr auf dem neuesten Stand. Sie können die Datei `holidays` aktualisieren.

Der Befehl `acctcon1` (gestartet mit dem Befehl `runacct`) sendet Mails an die Konten `root` und `adm`, sobald die Datei `/usr/lib/acct/holidays` nicht mehr auf dem neuesten Stand ist.

Aktualisieren Sie die nicht mehr auf dem neuesten Stand befindliche Datei `holidays`. Öffnen Sie dazu die Datei `/var/adm/acct/holidays` in einem Editor, und legen Sie darin die Haupt- und Nebenzeiten fest.

Als Hauptzeit wird die Periode betrachtet, in der Ihr System am aktivsten ist, z. B. an Arbeitstagen. Samstage und Sonntage sind ebenso wie die arbeitsfreien Tage, die Sie auflisten, für das Abrechnungssystem immer Nebenzeiten.

Die Datei `holidays` enthält drei Typen von Einträgen: Kommentare, das Jahr und die Periode mit den Hauptzeiten sowie eine Liste mit arbeitsfreien Tagen. Schauen Sie sich dazu das folgende Beispiel an:

\* Tabelle mit Haupt- und Nebenzeiten für das Abrechnungssystem

```
*
* Curr Prime Non-Prime
* Year Start Start
* 1992 0830 1700
*
* Day of Calendar Company
* Year Date Holiday
*
* 1 Jan 1 New Year's Day
* 20 Jan 20 Martin Luther King Day
* 46 Feb 15 President's Day
* 143 May 28 Memorial Day
* 186 Jul 3 4th of July
```

* 248	Sep 7	Labor Day
* 329	Nov 24	Thanksgiving
* 330	Nov 25	Friday after
* 359	Dec 24	Christmas Eve
* 360	Dec 25	Christmas Day
* 361	Dec 26	Day after Christmas

Die erste nicht auf Kommentar gesetzte Zeile muss das aktuelle Jahr (vier Stellen) sowie Beginn und Ende der Hauptzeit (auch jeweils vier Stellen) angeben. Das Konzept von Haupt- und Nebenzeiten wirkt sich lediglich auf die Art und Weise aus, in der die Abrechnungsprogramme die Abrechnungsdatensätze verarbeiten.

Wenn die Liste der arbeitsfreien Tage zu lang ist, generiert der Befehl **acctcon1** einen Fehler, und Sie müssen Ihre Liste kürzen. Mit maximal 20 arbeitsfreien Tagen sind Sie auf der sicheren Seite. Wenn Sie mehr arbeitsfreie Tage hinzufügen möchten, müssen Sie die Datei holidays jeden Monat anpassen.

## Abrechnungsdaten erfassen

Nach der Konfiguration der Systemabrechnung können Sie mit der Erfassung und Verarbeitung der verschiedenen Typen von Abrechnungsdaten beginnen.

### Systemdaten erfassen und zurückmelden:

Sie können das System so konfigurieren, dass es automatisch Daten erfasst und Berichte generiert.

Für die automatische Datenerfassung muss ein Mitglied der Gruppe adm als Abrechnungssystem konfiguriert sein. Die Konfiguration des Abrechnungssystems ermöglicht dem Dämon **cron**, die Befehle auszuführen, die Daten zu folgenden Aspekten generieren:

- Zeit, die jeder Benutzer angemeldet am System verbringt
- Nutzung von Verarbeitungseinheit, Speicher und E/A-Ressourcen
- Plattenspeicherplatz, den die Dateien jedes Benutzers belegen
- Nutzung von Druckern und Plottern
- Anzahl der Ausführung eines bestimmten Befehls

Das System schreibt einen Datensatz für jede Sitzung und jeden Prozess, die bzw. der abgeschlossen wird. Diese Datensätze werden in Gesamtabrechnungssätze (**tacct**) konvertiert, die vom Benutzer angeordnet und in einen täglichen Bericht eingefügt werden. In regelmäßigen Abständen werden die täglichen Berichte kombiniert, um Gesamtsummen für den definierten Abrechnungszeitraum zu erzeugen. Die Methoden für die Erfassung und Zurückmeldung der Daten sowie die verschiedenen Abrechnungsbefehle und -dateien werden in den folgenden Abschnitten beschrieben.

Obwohl die meisten Abrechnungsdaten automatisch erfasst und verarbeitet werden, kann ein Mitglied der Gruppe adm bestimmte Befehle über die Tastatur eingeben, um bestimmte Informationen zu erhalten.

### Zugehörige Tasks:

„Abrechnungssystem konfigurieren“ auf Seite 176

Sie können ein Abrechnungssystem konfigurieren.

### Zugehörige Verweise:

„Tastaturbefehle“ auf Seite 171

Ein Mitglied der Gruppe adm kann die folgenden Befehle über die Tastatur eingeben.

### Abrechnungsdaten zur Verbindungsdauer:

Daten zur Verbindungsdauer werden von den Befehlen **init** und **login** erfasst.

Wenn Sie sich anmelden, schreibt das Programm **login** einen Datensatz in die Datei `/etc/utmp`. Dieser Datensatz enthält Ihren Benutzernamen, Datum und Uhrzeit der Anmeldung und den Anmeldeport. Be-



fehler wie **who** verwenden diese Datei, um festzustellen, welche Benutzer an den verschiedenen Datensichtgeräten angemeldet sind. Wenn die Datei für die Abrechnung der Verbindungsdauer, `/var/adm/wtmp`, vorhanden ist, fügt der Befehl **login** eine Kopie dieses Anmeldedatensatzes zur Datei hinzu. Weitere Informationen finden Sie in den Beschreibungen der Befehle **init** und **login** in der Veröffentlichung "Commands Reference".

Wenn Ihr Anmeldeprogramm beendet wird (dies geschieht gewöhnlich, wenn Sie sich abmelden), protokolliert der Befehl **init** das Ende der Sitzung, indem er einen weiteren Datensatz in die Datei `/var/adm/wtmp` schreibt. Abmeldedatensätze unterscheiden sich insofern von Anmeldedatensätzen, dass sie für den Benutzernamen einen leeren Eintrag enthalten. Anmelde- und Abmeldedatensätze haben das in der Datei `utmp.h` beschriebene Format. Informationen zur Datei `utmp.h` finden Sie im Abschnitt `utmp.h`.

Der Befehl **acctwtmp** schreibt außerdem spezielle Einträge in die Datei `/var/adm/wtmp`, die sich auf Systemabschlüsse und Systemstarts beziehen.

#### **Zugehörige Konzepte:**

„Berichte zur Verbindungsdauer“ auf Seite 168

Zu Abrechnungsdatensätzen gehören Datensätze zu Anmeldungen, Abmeldungen, Systemabschlüssen und letzten Sitzungen.

#### **Prozessabrechnungsdaten:**

Das Abrechnungssystem erfasst Daten zur Ressourcennutzung für jeden Prozess während der Ausführung.

Diese Daten enthalten Folgendes:

- Benutzer- und Gruppennummer, unter der der Prozess ausgeführt wird
- Die ersten acht Zeichen des Befehlsnamens
- Numerischer 64-Bit-Schlüssel, der die Workload-Manager-Klasse enthält, zu der der Prozess gehört
- Abgelaufene Zeit und die vom Prozess verbrauchte Prozessorzeit
- Speicherbelegung
- Anzahl der übertragenen Zeichen
- Anzahl der für den Prozess gelesenen und geschriebenen Plattenblöcke

Der Befehl **accton** protokolliert diese Daten in einer angegebenen Datei, gewöhnlich der Datei `/var/adm/pacct`. Weitere Informationen finden Sie in der Beschreibung des Befehls **accton** in der Veröffentlichung "Commands Reference".

Zugehörige Befehle sind **startup**, **shutacct**, **dodisk**, **ckpacct** und **turnacct**. Weitere Informationen finden Sie in den Beschreibungen der Befehle **startup**, **shutacct**, **dodisk**, **ckpacct** und **turnacct** in der Veröffentlichung "Commands Reference".

#### **Zugehörige Konzepte:**

„Abrechnungsdatenberichte“ auf Seite 165

Nach der Erfassung der verschiedenen Typen von Abrechnungsdaten werden die Datensätze verarbeitet und in Berichte konvertiert.

#### **Prozessabrechnungsberichte:**

Es gibt zwei Befehle, die die abrechnungsbezogenen Daten verarbeiten, die in der Datei `/var/adm/pacct` oder einer anderen angegebenen Datei erfasst wurden.

Der Befehl **acctprc1** übersetzt die Benutzer-ID in einen Benutzernamen und schreibt ASCII-Datensätze, die gebührenpflichtige Einträge (CPU-Zeit zu Haupt- und Nebenzeiten, durchschnittliche Speichergröße und E/A-Daten) enthalten. Der Befehl **acctprc2** wandelt diese Datensätze in Gesamtabrechnungssätze um,

die vom Befehl **acctmrg** zu den täglichen Berichten hinzugefügt werden. Weitere Informationen finden Sie in der Beschreibung des Befehls **acctmrg** in der Veröffentlichung "Commands Reference".

Die Prozessabrechnungsdaten enthalten auch Informationen, die Sie für die Überwachung der Systemressourcennutzung verwenden können. Der Befehl **acctcms** fasst die Ressourcennutzung nach Befehlsnamen zusammen. Er liefert Informationen darüber, wie oft jeder Befehl ausgeführt wurde, wie viel Prozessorzeit und Speicher verwendet wurden und wie intensiv die Ressourcen verwendet wurden (auch Belastungsfaktor oder *Hog Factor* genannt). Der Befehl **acctcms** erzeugt Langzeitstatistiken zur Systemauslastung und liefert Informationen zur Gesamtsystemnutzung und zur Häufigkeit, mit der die Befehle verwendet wurden. Weitere Informationen finden Sie in der Beschreibung des Befehls **acctcms** in der Veröffentlichung "Commands Reference".

Der Befehl **acctcom** arbeitet mit denselben Daten wie der Befehl **acctcms**, stellt aber ausführliche Informationen zu jedem Prozess bereit. Sie können alle Prozessabrechnungssätze anzeigen oder gezielt Datensätze auswählen. Zu den Auswahlkriterien gehören die vom Prozess erzeugte Last, der Zeitraum, in dem der Prozess beendet wurde, der Name des Befehls, der Benutzer oder die Gruppe, der bzw. die den Prozess aufgerufen hat, der Name der WLM-Klasse, zu der der Prozess gehört, und der Port, an dem der Prozess ausgeführt wurde. Anders als andere Abrechnungsbefehle kann der Befehl **acctcom** von allen Benutzern ausgeführt werden. Weitere Informationen finden Sie in der Beschreibung des Befehls **acctcom** in der Veröffentlichung "Commands Reference".

#### **Abrechnungsdaten zur Plattenbelegung:**

Viele Abrechnungsdaten werden während des Verbrauchs der Ressourcen erfasst. Der Befehl **dodisk**, der nach den Angaben des Dämons **cron** ausgeführt wird, schreibt in regelmäßigen Abständen Datensätze zur Plattenbelegung für jeden Benutzer in die Datei `/var/adm/acct/nite(x)/dacct`.

Hierfür ruft der Befehl **dodisk** andere Befehle auf. Je nach Gründlichkeit der Abrechnungssuche kann der Befehl **diskusg** oder der Befehl **acctdusg** für die Erfassung der Daten verwendet werden. Der Befehl **acctdisk** wird verwendet, um einen Gesamtabrechnungssatz zu schreiben. Der Gesamtabrechnungssatz wiederum wird vom Befehl **acctmrg** verwendet, um den täglichen Abrechnungsbericht vorzubereiten.

Der Befehl **dodisk** berechnet einem Benutzer Gebühren für Dateien, zu denen im Anmeldeverzeichnis des Benutzers Verbindungen gefunden werden. Die Gebühren für jede Datei werden gleichmäßig auf die vorhandenen Verbindungen verteilt. Auf diese Weise werden die Kosten für die Verwendung einer Datei auf alle Benutzer verteilt, die die Datei verwenden. Wenn ein Benutzer eine Verbindung zu einer Datei aufgibt, entfallen für ihn die Gebühren. Weitere Informationen finden Sie in den Beschreibungen der Befehle **dodisk** und **cron** in der Veröffentlichung "Commands Reference".

#### **Zugehörige Konzepte:**

„Berichte zur Verbindungsdauer“ auf Seite 168

Zu Abrechnungsdatensätzen gehören Datensätze zu Anmeldungen, Abmeldungen, Systemabschlüssen und letzten Sitzungen.

#### **Abrechnungsdaten zur Druckerauslastung:**

Die Erfassung der Druckerauslastungsdaten ist eine Gemeinschaftsarbeit des Befehls **enq** und des Warteschlangendämons.

Der Befehl **enq** reiht die zu druckende Datei mit Benutzernamen, Jobnummer und Dateinamen in die Warteschlange ein. Nach dem Drucken der Datei schreibt der Befehl **qdaemon** einen ASCII-Datensatz in eine Datei, gewöhnlich die Datei `/var/adm/qacct`, der den Benutzernamen, die Benutzernummer und die Anzahl der gedruckten Seiten enthält. Sie können diese Datensätze sortieren und in Gesamtabrechnungssätze konvertieren. Weitere Informationen finden Sie in den Beschreibungen der Befehle **enq** und **qdaemon** in der Veröffentlichung "Commands Reference".

#### **Zugehörige Konzepte:**

„Abrechnungsbericht zur Druckerauslastung“ auf Seite 169

Der ASCII-Datensatz in der Datei /var/adm/qacct kann in einen Gesamtabrechnungssatz konvertiert werden, der vom Befehl **acctmerg** dem täglichen Bericht hinzugefügt wird.

### Gebührenabrechnungsdaten:

Sie können einen Gesamtabrechnungssatz im ASCII-Format in der Datei /var/adm/fee erzeugen.

Mit dem Befehl **chargefee** können Sie einen Gesamtabrechnungssatz im ASCII-Format in der Datei /var/adm/fee erzeugen. Diese Datei wird vom Befehl **acctmerg** zu den täglichen Berichten hinzugefügt.

Weitere Informationen finden Sie in den Beschreibungen der Befehle **chargefee** und **acctmerg** in der Veröffentlichung "Commands Reference".

### Zugehörige Konzepte:

„Gebührenabrechnungsbericht“ auf Seite 169

Wenn Sie den Befehl **chargefee** verwendet haben, um Benutzern Gebühren für Services, wie z. B. Datei-wiederherstellungen, Consulting oder Arbeitsmaterialien, zu berechnen, wird ein Gesamtabrechnungssatz in die Datei /var/adm/fee geschrieben. Diese Datei wird vom Befehl **acctmerg** zu den täglichen Berichten hinzugefügt.

## Fehlerbehebung bei der Systemabrechnung

Verwenden Sie die folgenden Methoden für die Behebung einiger der grundlegenden Probleme, die bei der Systemabrechnung auftreten können. Wenn Ihr Problem in den verfügbaren Fehlerbehebungsinformationen nicht adressiert wird, wenden Sie sich an Ihren IBM Ansprechpartner.

### tacct-Fehler beheben:

Wenn Sie das Abrechnungssystem verwenden, um Benutzern Gebühren für Systemressourcen zu berechnen, ist die Datei /var/adm/acct/sum/tacct sehr wichtig. Gelegentlich tauchen mysteriöse **tacct**-Datensätze auf, die negative Zahlen, doppelte Benutzernummern oder die Benutzernummer 65.535 enthalten. Diese Probleme können behoben werden.

Sie müssen Root- oder Administratorgruppenberechtigung haben.

Zum Korrigieren einer **tacct**-Datei führen Sie die folgenden Schritte aus:

1. Wechseln Sie mit dem folgenden Befehl in das Verzeichnis /var/adm/acct/sum:

```
cd /var/adm/acct/sum
```

2. Verwenden Sie den Befehl **prtacct**, um die Gesamtabrechnungsdatei **tacctprev** zu prüfen. Geben Sie Folgendes ein:

```
prtacct tacctprev
```

Der Befehl **prtacct** formatiert und zeigt die Datei **tacctprev** an, so dass Sie Verbindungsdauer, Verarbeitungszeit, Plattenbelegung und Druckerauslastung prüfen können.

3. Wenn die Datei **tacctprev** korrekt zu sein scheint, konvertieren Sie die letzte Datei **tacct** **.mmtt** von einer Binärdatei in eine ASCII-Datei. Im folgenden Beispiel konvertiert der Befehl **acctmerg** die Datei **tacct.mmmtt** in eine ASCII-Datei mit dem Namen **tacct.neu**:

```
acctmerg -v < tacct.mmmtt > tacct.neu
```

**Anmerkung:** Der Befehl **acctmerg** erzeugt zusammen mit dem Flag **-a** ebenfalls eine ASCII-Ausgabe. Das Flag **-v** erzeugt eine genauere Schreibweise für Gleitkommazahlen.

Der Befehl **acctmerg** wird verwendet, um die temporären Abrechnungsdatensatzberichte zu einem kumulativen Gesamtbericht (**tacct**) zusammenzuführen. Dieser kumulative Gesamtbericht ist die Quelle,

aus der der Befehl **monacct** den Monatsübersichtsbericht im ASCII-Format erzeugt. Da die Befehlsprozedur **monacct** alle Dateien `tacct.mmtt` entfernt, erstellen Sie die `tacct`-Datei erneut, indem Sie diese Dateien zusammenführen.

4. Öffnen Sie die Datei `tacct.neu` in einem Editor, entfernen Sie die ungültigen Datensätze, und schreiben Sie die Datensätze mit den doppelt vorhandenen Benutzernummern in eine andere Datei. Geben Sie Folgendes ein:

```
acctmerg -i < tacct.neu > tacct.mmtt
```

5. Erstellen Sie die Datei `tacct` mit dem folgenden Befehl erneut:

```
acctmerg tacctprev < tacct.mmtt > tacct
```

### **wtmp-Fehler beheben:**

Die Datei `/var/adm/wtmp` bzw. "who temp" kann im täglichen Betrieb des Abrechnungssystems zu Problemen führen. Sie können `wtmp`-Fehler beheben.

Zur Ausführung dieser Prozedur müssen Sie Root- oder Administratorgruppenberechtigung haben.

Wenn das Datum geändert wird und das System im Mehrbenutzermodus betrieben wird, werden Datensätze zu Datumsänderungen in die Datei `/var/adm/wtmp` geschrieben. Wenn eine Datumsänderung festgestellt wird, passt der Befehl **wtmpfix** die Zeitmarken in den `wtmp`-Datensätzen an. Einige Kombinationen von Datumsänderungen und Systemwiederanläufen können dem Befehl **wtmpfix** entgehen und deshalb dazu führen, dass der Befehl **acctcon1** fehlschlägt und der Befehl **runacct** Mails an die Konten **root** und **adm** sendet, in denen ungültige Datumsangaben aufgelistet sind.

Gehen Sie zum Beheben von `wtmp`-Fehlern wie folgt vor:

1. Wechseln Sie mit dem folgenden Befehl in das Verzeichnis `/var/adm/acct/nite`:

```
cd /var/adm/acct/nite
```

2. Konvertieren Sie die Binärdatei `wtmp` mit dem folgenden Befehl in eine editierbare ASCII-Datei:

```
fwtmp < wtmp.mmtt > wtmp.neu
```

Der Befehl **fwtmp** konvertiert die Datei `wtmp` aus dem Binärformat in das ASCII-Format.

3. Öffnen Sie die ASCII-Datei `wtmp.neu` in einem Editor, und löschen Sie die beschädigten Datensätze bzw. alle Datensätze vom Anfang der Datei bis hin zur erforderlichen Datumsänderung. Geben Sie dazu Folgendes ein:

```
vi wtmp.neu
```

4. Konvertieren Sie die ASCII-Datei `wtmp.neu` mit dem folgenden Befehl zurück in das Binärformat:

```
fwtmp -ic < wtmp.neu > wtmp.mmtt
```

5. Falls sich die Datei `wtmp` nicht mehr reparieren lässt, erstellen Sie mit dem Befehl **nulladm** eine leere Datei `wtmp`. Dies sorgt dafür, dass für die Verbindungen keinerlei Gebühren erhoben werden.

```
nulladm wtmp
```

Der Befehl **nulladm** erstellt die angegebene Datei mit Lese- und Schreibberechtigungen für den Dateieigner und die Gruppe sowie Leseberechtigungen für andere Benutzer. Er stellt sicher, dass **adm** Dateieigner und Gruppe ist.

### **Zugehörige Tasks:**

„Abrechnungsfehler beheben“ auf Seite 189

Sie können Datums- und Zeitmarkeninkonsistenzen korrigieren.

### **Ungültige Berechtigungen für Abrechnungsdateien korrigieren:**

Für die Verwendung des Abrechnungssystems müssen Dateieigentumsrecht und Dateiberechtigungen korrekt sein.

Zur Ausführung dieser Prozedur müssen Sie Root- oder Administratorgruppenberechtigung haben.

Das Verwaltungskonto **adm** ist Eigner des Abrechnungsbefehls und der Abrechnungsscripts. Die einzige Ausnahme ist `/var/adm/acct/accton`, dessen Eigner Root ist.

Zum Korrigieren ungültiger Berechtigungen für Abrechnungsdateien gehen Sie wie folgt vor:

1. Geben Sie Folgendes ein, um die Dateiberechtigungen mit dem Befehl **ls** zu überprüfen.

```
ls -l /var/adm/acct
```

```
-rws--x--- 1 adm adm 14628 Mar 19 08:11 /var/adm/acct/fiscal
-rws--x--- 1 adm adm 14628 Mar 19 08:11 /var/adm/acct/nite
-rws--x--- 1 adm adm 14628 Mar 19 08:11 /var/adm/acct/sum
```

2. Passen Sie die Dateiberechtigungen bei Bedarf mit dem Befehl **chown** an. Die Berechtigungen sind 755 (alle Berechtigungen für den Eigner und Lese- und Ausführungsrechte für alle anderen). Außerdem muss das Verzeichnis selbst vor Schreibzugriffen durch andere geschützt werden. Beispiel:

- a. Wechseln Sie mit dem folgenden Befehl in das Verzeichnis `/var/adm/acct`:

```
cd /var/adm/acct
```

- b. Ändern Sie die Eigentümergruppe für die Verzeichnisse `sum`, `nite` und `fiscal` mit dem folgenden Befehl in **adm**:

```
chown adm sum/* nite/* fiscal/*
```

Um Manipulationen von Benutzern vorzubeugen, die Gebühren umgehen möchten, verweigern Sie anderen den Schreibzugriff auf diese Dateien. Ändern Sie den Gruppeneigner für **accton** in **adm** und die Berechtigungen in 710, d. h. keine Berechtigungen für andere. Prozesse, deren Eigner **adm** ist, können den Befehl **accton** ausführen, Standardbenutzer jedoch nicht.

3. Eigner der Datei `/var/adm/wtmp` muss ebenfalls **adm** sein. Wenn Root Eigner von `/var/adm/wtmp` ist, wird während des Systemstarts die folgende Nachricht angezeigt:

```
/var/adm/acct/startup: /var/adm/wtmp: Permission denied
```

Wenn Sie das Eigentumsrecht für `/var/adm/wtmp` korrigieren möchten, ändern Sie es mit dem folgenden Befehl in die Gruppe **adm**:

```
chown adm /var/adm/wtmp
```

### Abrechnungsfehler beheben:

Sie können Datums- und Zeitmarkeninkonsistenzen korrigieren.

Zur Ausführung dieser Prozedur müssen Sie Root- oder Administratorgruppenberechtigung haben.

Bei der Verarbeitung der Datei `/var/adm/wtmp` können Warnungen erzeugt werden, die per Mail an Root gesendet werden. Die Datei `wtmp` enthält Informationen, die von `/etc/init` und `/bin/login` erfasst werden und von Abrechnungsscripts hauptsächlich für die Berechnung der Verbindungsdauer (der Zeit, die ein Benutzer angemeldet ist) verwendet werden. Leider sorgen Datumsänderungen für Verwirrung in dem Programm, das die Datei `wtmp` verarbeitet. Aus diesem Grund sendet der Befehl **runacct** Mails an `root` und `adm`, in denen er Fehler meldet, die nach einer Datumsänderung seit der letzten Durchführung der Abrechnung auftreten.

1. Stellen Sie fest, ob Sie Fehlernachrichten erhalten haben. Der Befehl **acctcon1** gibt Fehlernachrichten aus, die der Befehl **runacct** per Mail an `root` und `adm` sendet. Wenn der Befehl **acctcon1** nach einer Datumsänderung die Zeiten für die Verbindungsdauer nicht erfassen kann, kann `adm` eine Mail-Nachricht wie die folgende erhalten:

```
Mon Jan 6 11:58:40 CST 1992
acctcon1: bad times: old: Tue Jan 7 00:57:14 1992
new: Mon Jan 6 11:57:59 1992
```

```
acctcon1: bad times: old: Tue Jan 7 00:57:14 1992
new: Mon Jan 6 11:57:59 1992
acctcon1: bad times: old: Tue Jan 7 00:57:14 1992
new: Mon Jan 6 11:57:59 1992
```

2. Passen Sie die Datei `wtmp` mit dem folgenden Befehl an:

```
/usr/sbin/acct/wtmpfix wtmp
```

Der Befehl **wtmpfix** untersucht die Datei `wtmp` auf Datums- und Zeitmarkeninkonsistenzen und behebt Probleme, die zum Scheitern des Befehls **acctcon1** führen könnten. Einige Datumsänderungen entgehen dem Befehl **wtmpfix** jedoch.

3. Führen Sie die Abrechnung direkt vor dem Systemabschluss oder direkt nach dem Systemstart durch. Wenn Sie den Befehl **runacct** zu diesen Zeiten ausführen, verringert sich die Anzahl der Einträge mit ungültigen Zeitangaben. Der Befehl **runacct** sendet so lange Mails an die Konten `root` und `adm`, bis Sie das Script **runacct** in einem Editor öffnen, den Abschnitt `WTMPFIX` suchen und die Zeile auf Kommentar setzen, die dafür sorgt, dass das Dateiprotokoll per Mail an die Konten `root` und `adm` gesendet wird.

### Zugehörige Tasks:

„wtmp-Fehler beheben“ auf Seite 188

Die Datei `/var/adm/wtmp` bzw. "who temp" kann im täglichen Betrieb des Abrechnungssystems zu Problemen führen. Sie können `wtmp`-Fehler beheben.

*Abrechnungsfehler beim Ausführen des Befehls runacct:*

Bei der Ausführung des Befehls **runacct** können Fehler auftreten.

**Anmerkung:** Für die Ausführung des Befehls **runacct** müssen Sie Rootberechtigung besitzen oder Mitglied der Gruppe `adm` sein.

Häufig verarbeitet der Befehl **runacct** sehr große Dateien. Die Prozedur umfasst mehrere Durchläufe durch bestimmte Dateien und verbraucht erhebliche Systemressourcen. Da der Befehl **runacct** erhebliche Ressourcen verbraucht, wird er gewöhnlich am frühen Morgen ausgeführt, wenn er die Maschine zur vollen Verfügung hat und niemanden stört.

Der Befehl **runacct** ist ein Script, das in mehrere Abschnitte aufgeteilt ist. Die Abschnitte ermöglichen Ihnen, den Befehl dort erneut zu starten, wo er gestoppt wurde, ohne das vollständige Script erneut starten zu müssen.

Wenn der Befehl **runacct** Probleme feststellt, sendet er je nachdem, wo der Fehler aufgetreten ist, Fehlernachrichten an verschiedene Ziele. Normalerweise sendet er ein Datum und eine Nachricht an die Konsole und weist Sie darin an, in der Datei `activeMMTT` (z. B. `active0621` für den 21. Juni) im Verzeichnis `/usr/adm/acct/nite` nachzusehen. Wenn der Befehl **runacct** abgebrochen wird, verschiebt er die vollständige Datei `active` nach `activeMMTT` und fügt eine Nachricht an, in der das Problem beschrieben wird.

In den folgenden Tabellen sind Fehlernachrichten beschrieben, die bei der Ausführung des Befehls **runacct** auftreten können.

### Anmerkung:

- Die Abkürzung `MMTT` steht für Monat und Tag, z. B. `0102` für den 2. Januar. Wenn beispielsweise im Prozess `CONNECT1` am 2. Januar ein nicht behebbarer Fehler auftritt, wird die Datei `active0102` mit der Fehlernachricht erstellt.
- Die Abkürzung "SE message" steht für die Standardfehlernachricht, z. B.:

```
***** ACCT ERRORS : see active0102 *****
```

### Vorläufige Status und Fehlermeldungen des Befehls **runacct**

Status	Befehl	Nicht behebbar?	Fehlermeldung	Ziele
pre	<b>runacct</b>	ja	* 2 CRONS or ACCT PROBLEMS * ERROR: locks found, run aborted	console, mail, active
pre	<b>runacct</b>	ja	runacct: Insufficient space in /usr ( <i>nnn</i> blks); Terminating procedure	console, mail, active
pre	<b>runacct</b>	ja	SE message; ERROR: acctg already run for 'date': check lastdate	console, mail, activeMMTT
pre	<b>runacct</b>	nein	* SYSTEM ACCOUNTING STARTED *	console
pre	<b>runacct</b>	nein	restarting acctg for 'date' at STATE	console active, console
pre	<b>runacct</b>	nein	restarting acctg for 'date' at state (argument \$2) previous state was STATE	active
pre	<b>runacct</b>	ja	SE message; Error: runacct called with invalid arguments	console, mail, activeMMTT

### Status und Fehlermeldungen des Befehls **runacct**

Status	Befehl	Nicht behebbar?	Fehlermeldung	Ziele
SETUP	<b>runacct</b>	nein	ls -l fee pacct* /var/adm/wtmp	active
SETUP	<b>runacct</b>	ja	SE message; ERROR: turnacct switch returned rc=error	console, mail, activeMMTT
SETUP	<b>runacct</b>	ja	SE message; ERROR: SpacctMMTT already exists file setups probably already run	activeMMTT
SETUP	<b>runacct</b>	ja	SE message; ERROR: wtmpMMTT already exists: run setup manually	console, mail, activeMMTT
WTMPFIX	<b>wtmpfix</b>	nein	SE message; ERROR: wtmpfix errors see xtmperrorMMTT	activeMMTT, wtmperrorMMTT
WTMPFIX	<b>wtmpfix</b>	nein	wtmp processing complete	active
CONNECT1	<b>acctcon1</b>	nein	SE message; (errors from acctcon1 log)	console, mail, activeMMTT
CONNECT2	<b>acctcon2</b>	nein	connect acctg complete	active
PROCESS	<b>runacct</b>	nein	WARNING: accounting already run for pacctN	active
PROCESS	<b>acctprc1 acctprc2</b>	nein	process acctg complete for SpacctNMMTT	active
PROCESS	<b>runacct</b>	nein	all process actg complete for date	active
MERGE	<b>acctmerge</b>	nein	tacct merge to create dayacct complete	active

### Status und Fehlermeldungen des Befehls **runacct**

Status	Befehl	Nicht behebbar?	Fehlermeldung	Ziele
FEES	<b>acctmerg</b>	nein	merged fees OR no fees	active
DISK	<b>acctmerg</b>	nein	merged disk records OR no disk records	active
MERGEACCT	<b>acctmerg</b>	nein	WARNING: recreating sum/tacct	active
MERGEACCT	<b>acctmerg</b>	nein	updated sum/tacct	active
CMS	<b>runacct</b>	nein	WARNING: recreating sum/cms	active
CMS	<b>acctcms</b>	nein	command summaries complete	active
CLEANUP	<b>runacct</b>	nein	system accounting completed at 'date'	active
CLEANUP	<b>runacct</b>	nein	*SYSTEM ACCOUNTING COMPLETED*	console
<wrong>	<b>runacct</b>	ja	SE message; ERROR: invalid state, check STATE	console, mail, activeMMTT

**Anmerkung:** Die Angabe <wrong> in der vorherigen Tabelle ist kein Status, sondern gibt an, dass ein ungültiger Status in die Statusdatei /usr/adm/acct/nite/statefile geschrieben wurde.

#### Zusammenfassung der Nachrichtenziele

Ziel	Beschreibung
console	Einheit /dev/console
mail	Nachricht, die per Mail an die Konten <b>root</b> und <b>adm</b> gesendet wird
active	Datei /usr/adm/acct/nite/active
activeMMTT	Datei /usr/adm/acct/nite/activeMMTT
wtmperrMMTT	Datei /usr/adm/acct/nite/wtmperrorMMTT
STATE	Aktueller Status in der Datei /usr/adm/acct/nite/statefile
fd2log	Alle anderen Fehlermeldungen

## System Resource Controller

Der System Resource Controller (SRC) stellt eine Reihe von Befehlen und Subroutinen zur Verfügung, die dem Systemmanager und dem Programmierer die Erstellung und Steuerung von Subsystemen erleichtern.

*Subsysteme* sind Programme oder Prozesse bzw. Programmgruppen oder Prozessgruppen, die in der Regel in der Lage sind, unabhängig voneinander oder mit einem steuernden System ausgeführt zu werden. Ein Subsystem ist als Einheit konzipiert, die eine bestimmte Funktion bereitstellt.

Der SRC wurde so gestaltet, dass der Bedarf an Bedieneingriffen minimiert wird. Er ist ein Mechanismus für die Steuerung von Subsystemprozessen über eine allgemeine Befehlszeile oder C-Schnittstelle. Dieser Mechanismus umfasst Folgendes:

- eine konsistente Schnittstelle für Start-, Stopp- und Statusabfragen,
- Protokollierung abnormaler Beendigungen von Subsystemen,
- Benachrichtigungsprogramm, das bei der abnormalen Systembeendigung zugehöriger Prozesse aufgerufen wird,
- Erstellung von Traces für Subsysteme, Subsystemgruppen oder Subserver,
- Unterstützung für die Steuerung von Operationen auf einem fernen System,



- Aktualisierung eines Subsystems (z. B. nach der Änderung von Konfigurationsdaten).

Der SRC ist hilfreich, wenn Sie eine allgemeine Methode für das Starten und Stoppen von Prozessen sowie das Erfassen von Statusinformationen zu Prozessen verwenden möchten.

#### Zugehörige Konzepte:

„Einführung in AIX für BSD-Systemmanager“ auf Seite 342

Die folgenden Tipps sind für BSD-Systemmanager (Berkeley Software Distribution) als Einführung in die Verwaltung von AIX bestimmt.

## Subsystemkomponenten

Im Folgenden werden die Eigenschaften und Komponenten eines Subsystems beschrieben.

Ein Subsystem kann eine oder mehrere der folgenden Eigenschaften haben:

- Es ist dem System unter seinem Namen bekannt.
- Es erfordert eine komplexere Ausführungsumgebung als eine Subroutine oder ein nicht privilegiertes Programm.
- Es enthält Anwendungsprogramme und Bibliotheken sowie Subsystemcode.
- Es steuert Ressourcen, die nach Namen gestartet oder gestoppt werden können.
- Es muss benachrichtigt werden, wenn ein zugehöriger Prozess Ressourcen nicht bereinigen oder wiederherstellen kann.
- Es erfordert mehr Betriebssteuerung als ein einfacher Dämonprozess.
- Es muss von einem fernen Bediener gesteuert werden.
- Es implementiert Subserver für die Verwaltung bestimmter Ressourcen.
- Es verlagert sich nicht selbst in den Hintergrund.

Einige Beispiele für Subsysteme sind ypserv, ntsd, qdaemon, inetd, syslogd und sendmail.

**Anmerkung:** Einzelheiten zu den SRC-Funktionen finden Sie in der Beschreibung jedes einzelnen Subsystems.

Verwenden Sie den Befehl **lssrc -a**, um aktive und inaktive Subsysteme auf Ihrem System aufzulisten.

Im Folgenden finden Sie die Definitionen für Subsystemgruppen und Subserver:

#### Subsystemgruppe

Eine *Subsystemgruppe* ist eine Gruppe bestimmter Subsysteme. Wenn Sie Subsysteme gruppieren, können mehrere Subsysteme gleichzeitig gesteuert werden. Beispiele für Subsystemgruppen sind TCP/IP, SNA Services, Network Information System (NIS) und Network File Systems (NFS).

#### Subserver

Ein *Subserver* ist ein Programm oder Prozess, der zu einem Subsystem gehört. Ein Subsystem kann mehrere Subserver haben und ist für das Starten, Stoppen und die Bereitstellung des Status von Subservern zuständig. Subserver können nur für Subsysteme, die über IPC-Nachrichtenwarteschlangen und -Sockets kommunizieren, definiert werden. Subsysteme, die die Kommunikation über Signale verwenden, unterstützen keine Subserver.

Subserver werden gestartet, wenn ihre übergeordneten Subsysteme gestartet werden. Wenn Sie versuchen, einen Subserver zu starten, und das übergeordnete Subsystem nicht aktiv ist, startet der Befehl **startsrc** auch das Subsystem.

## SRC-Hierarchie

Die SRC-Hierarchie (System Resource Controller) beginnt mit dem Betriebssystem, gefolgt von einer Subsystemgruppe (z. B. **tcpip**), die ein Subsystem enthält (z. B. den Dämon **inetd**), das wiederum mehrere Subserver besitzen kann (z. B. den Dämon **ftp** und den Befehl **finger**).

## SRC-Verwaltungsbefehle

Sie können den System Resource Controller (SRC) über die Befehlszeile verwalten.

Im Folgenden finden Sie eine Liste mit den Verwaltungsbefehlen für den SRC:

Eintrag	Beschreibung
Dämon <b>srcmstr</b>	Startet den System Resource Controller.
Befehl <b>startsrc</b>	Startet ein Subsystem, eine Subsystemgruppe oder einen Subserver.
Befehl <b>stopsrc</b>	Stoppt ein Subsystem, eine Subsystemgruppe oder einen Subserver.
Befehl <b>refresh</b>	Aktualisiert ein Subsystem.
Befehl <b>traceson</b>	Aktiviert die Tracerstellung für ein Subsystem, eine Subsystemgruppe oder einen Subserver.
Befehl <b>tracesoff</b>	Inaktiviert die Tracerstellung für ein Subsystem, eine Subsystemgruppe oder einen Subserver.
Befehl <b>lssrc</b>	Ruft den Status eines Subsystems ab.

## System Resource Controller starten

Der System Resource Controller (SRC) wird während der Systeminitialisierung über einen Datensatz für den Dämon `/usr/sbin/srcmstr` in der Datei `/etc/inittab` gestartet.

Im Folgenden sind die Voraussetzungen für das Starten des SRC beschrieben:

- Sie benötigen Rootberechtigung, um die Datei `/etc/inittab` lesen und schreiben zu können.
- Sie benötigen Rootberechtigung, um den Befehl **mkitab** auszuführen.
- Der Datensatz für den Dämon **srcmstr** muss in der Datei `/etc/inittab` enthalten sein.

Die Standarddatei `/etc/inittab` enthält bereits einen solchen Datensatz, somit ist diese Prozedur unter Umständen nicht erforderlich. Sie können den SRC auch über die Befehlszeile, ein Profil oder ein Shell-Script ausführen, aber es gibt verschiedene Gründe, die dafür sprechen, den SRC während der Initialisierung zu starten:

- Wenn Sie den SRC über die Datei `/etc/inittab` starten, kann der Befehl **init** den SRC erneut starten, falls er aus irgendeinem Grund gestoppt wird.
- Der SRC vereinfacht und verringert die erforderlichen Bedienereingriffe für die Steuerung von Subsystemen. Wenn Sie den SRC über eine andere Quelle als die Datei `/etc/inittab` starten, ist dies kontraproduktiv.
- Die Standarddatei `/etc/inittab` enthält einen Datensatz für das Starten des Subsystems für die Druckplanung (**qdaemon**) mit dem Befehl **startsrc**. Standardinstallationen haben andere Subsysteme, die ebenfalls mit **startsrc**-Befehlen in der Datei `/etc/inittab` gestartet werden. Da der Befehl **srcmstr** voraussetzt, dass der SRC aktiv ist, scheitern diese **startsrc**-Befehle, wenn Sie den Dämon **srcmstr** aus der Datei `/etc/inittab` entfernen.

**Anmerkung:** Diese Prozedur ist nur erforderlich, wenn die Datei `/etc/inittab` noch keinen Datensatz für den Dämon **srcmstr** enthält.

1. Erstellen Sie mit dem Befehl **mkitab** einen Datensatz für den Dämon **srcmstr** in der Datei `/etc/inittab`. Wenn Sie beispielsweise denselben Datensatz wie in der Standarddatei `/etc/inittab` erstellen möchten, geben Sie Folgendes ein:

```
mkitab -i fbcheck srcmstr:2:respawn:/usr/sbin/srcmstr
```

Das Flag **-i fbcheck** stellt sicher, dass der Datensatz vor allen Subsystemdatensätzen eingefügt wird.

2. Geben Sie Folgendes ein, um den Befehl **init** anzuweisen, die Datei `/etc/inittab` erneut zu verarbeiten:

```
telinit q
```

Wenn der Befehl **init** die Datei `/etc/inittab` erneut bearbeitet, verarbeitet er den neu eingefügten Datensatz für den Dämon **srcmstr** und startet den SRC.

**Zugehörige Konzepte:**

„Subsystemsteuerung“ auf Seite 196

Die Tracerstellung für eine SRC-Ressource (System Resource Controller), z. B. ein Subsystem, eine Subsystemgruppe oder einen Subserver, kann mit dem Befehl **traceson** aktiviert und mit dem Befehl **traceoff** inaktiviert werden.

#### Zugehörige Tasks:

„Subsystem oder Subsystemgruppe aktualisieren“ auf Seite 196

Verwenden Sie den Befehl **refresh**, um eine SRC-Ressource (System Resource Controller), z. B. ein Subsystem oder eine Subsystemgruppe, anzuweisen, sich selbst zu aktualisieren.

### Subsystem, Subsystemgruppe oder Subserver starten und stoppen

Verwenden Sie den Befehl **startsrc**, um eine SRC-Ressource (System Resource Controller), z. B. ein Subsystem, eine Subsystemgruppe oder einen Subserver, zu starten. Mit dem Befehl **stopsrc** können Sie eine SRC-Ressource, z. B. ein Subsystem, eine Subsystemgruppe oder einen Subserver, stoppen.

Im Folgenden sind die Voraussetzungen für das Starten oder Stoppen eines Subsystems, einer Subsystemgruppe oder Subservers beschrieben:

- Wenn Sie eine SRC-Ressource starten oder stoppen möchten, muss der SRC aktiv sein. Der SRC wird normalerweise während der Systeminitialisierung gestartet. Die Standarddatei `/etc/inittab`, die bestimmt, welche Prozesse während der Initialisierung gestartet werden, enthält einen Datensatz für den Dämon **srcmstr** (SRC). Um festzustellen, ob der SRC aktiv ist, geben Sie `ps -A` ein. Suchen Sie in der Ausgabe dieses Befehls einen Prozess mit dem Namen `srcmstr`.
- Der Benutzer bzw. Prozess, der eine SRC-Ressource startet, muss Rootberechtigung besitzen. Der Prozess, der das System initialisiert (Befehl **init**), hat Rootberechtigung.
- Der Benutzer bzw. Prozess, der eine SRC-Ressource stoppt, muss Rootberechtigung besitzen.

Der Befehl **startsrc** kann wie folgt verwendet werden:

- In der Datei `/etc/inittab`, so dass die Ressource während der Systeminitialisierung gestartet wird
- In der Befehlszeile
- Mit SMIT

Wenn Sie eine Subsystemgruppe starten, werden auch alle zugehörigen Subsysteme gestartet. Wenn Sie ein Subsystem starten, werden auch alle zugehörigen Subserver gestartet. Wenn Sie einen Subserver starten, wird auch das übergeordnete Subsystem gestartet, falls dieses noch nicht aktiv ist.

Wenn Sie ein Subsystem stoppen, werden auch alle Subserver gestoppt. Durch das Stoppen eines Subservers ändert sich jedoch nicht der Status des übergeordneten Subsystems.

Die Befehle **startsrc** und **stopsrc** enthalten Flags, mit denen Anforderungen an lokale und ferne Hosts abgesetzt werden können. Informationen zu den Konfigurationsvoraussetzungen für die Unterstützung fernere SRC-Anforderungen finden Sie in der Beschreibung des Befehls **srcmstr**.

#### Subsystemtasks starten und stoppen

<i>Task</i>	<i>SMIT-Direktaufruf</i>	<i>Befehl oder Datei</i>
Subsystem starten	<b>smit startsys</b>	<code>/bin/startsrc -s Subsystemname</code> (oder <code>/etc/inittab</code> bearbeiten)
Subsystem stoppen	<b>smit stopsys</b>	<code>/bin/stopsrc -s Subsystemname</code>

#### Zugehörige Informationen:

stopsrc command

startsrc command

srcmstr command

## Status von Subsystemen anzeigen

Verwenden Sie den Befehl **lssrc**, um den Status einer SRC-Ressource (System Resource Controller), z. B. eines Subsystems, einer Subsystemgruppe oder eines Subservers, anzuzeigen.

Alle Subsysteme können einen kurzen Statusbericht zurückgeben, der angibt, zu welcher Gruppe das Subsystem gehört, ob das Subsystem aktiv ist und welche Prozess-ID (PID) das Subsystem hat. Wenn ein Subsystem nicht das Kommunikationsverfahren über Signale verwendet, kann es so programmiert werden, dass es einen ausführlichen Bericht zurückgibt, der zusätzliche Statusinformationen enthält.

Der Befehl **lssrc** besitzt Flags und Parameter für die Angabe des Subsystems mit dem Namen oder mit der PID, für das Auflisten aller Subsysteme, für das Anfordern eines kurzen oder ausführlichen Berichts und für das Anfordern des Status von SRC-Ressourcen auf dem lokalen oder auf fernen Hosts.

Informationen zu den Konfigurationsvoraussetzungen für die Unterstützung ferner SRC-Anforderungen finden Sie in der Beschreibung des Befehls **srcmstr**.

Tasks zum Anzeigen des Status von Subsystemen

<i>Task</i>	<i>SMIT-Direktaufruf</i>	<i>Befehl oder Datei</i>
Status eines Subsystems anzeigen (Langformat)	<b>smit qssys</b>	<b>lssrc -l -s Subsystemname</b>
Status aller Subsysteme anzeigen	<b>smit lsssys</b>	<b>lssrc -a</b>
Status aller Subsysteme auf einem bestimmten Host anzeigen		<b>lssrc -hHostname -a</b>

## Subsystem oder Subsystemgruppe aktualisieren

Verwenden Sie den Befehl **refresh**, um eine SRC-Ressource (System Resource Controller), z. B. ein Subsystem oder eine Subsystemgruppe, anzuweisen, sich selbst zu aktualisieren.

Im Folgenden sind die Voraussetzungen für die Aktualisierung eines Subsystems oder einer Subsystemgruppe beschrieben:

- Der SRC muss aktiv sein.
- Die Ressource, die Sie aktualisieren möchten, darf nicht das Kommunikationsverfahren über Signale verwenden.
- Die Ressource, die Sie aktualisieren möchten, muss so programmiert sein, dass sie auf die Aktualisierungsanforderung reagiert.

Der Befehl **refresh** hat Flags und Parameter, mit denen Sie das Subsystem mit Namen oder PID angeben können. Sie können mit diesem Befehl auch die Aktualisierung eines Subsystems oder einer Subsystemgruppe auf dem lokalen System oder auf fernen Hosts anfordern. Informationen zu den Konfigurationsvoraussetzungen für die Unterstützung ferner SRC-Anforderungen finden Sie in der Beschreibung des Befehls **srcmstr**.

Subsystem oder Subsystemgruppe aktualisieren

<b>Task</b>	<b>SMIT-Direktaufruf</b>	<b>Befehl oder Datei</b>
Ein Subsystem aktualisieren	<b>smit refresh</b>	<b>refresh -s Subsystem</b>

### Zugehörige Tasks:

„System Resource Controller starten“ auf Seite 194

Der System Resource Controller (SRC) wird während der Systeminitialisierung über einen Datensatz für den Dämon `/usr/sbin/srcmstr` in der Datei `/etc/inittab` gestartet.

## Subsystemsteuerung

Die Tracerstellung für eine SRC-Ressource (System Resource Controller), z. B. ein Subsystem, eine Subsystemgruppe oder einen Subserver, kann mit dem Befehl **traceson** aktiviert und mit dem Befehl **traceoff** inaktiviert werden.

Mit dem Befehl **traceson** können Sie die Traceerstellung für eine SRC-Ressource, z. B. ein Subsystem, eine Subsystemgruppe oder einen Subserver, aktivieren.

Mit dem Befehl **tracesoff** können Sie die Traceerstellung für eine SRC-Ressource, z. B. ein Subsystem, eine Subsystemgruppe oder einen Subserver, inaktivieren.

Die Befehle **traceson** und **tracesoff** können verwendet werden, um über Remote-Zugriff die Traceerstellung auf einem bestimmten Host zu aktivieren bzw. zu inaktivieren. Informationen zu den Konfigurationsvoraussetzungen für die Unterstützung ferner SRC-Anforderungen finden Sie in der Beschreibung des Befehls **srcmstr**.

Voraussetzungen

- Wenn Sie die Traceerstellung für eine SRC-Ressource aktivieren oder inaktivieren möchten, muss der SRC aktiv sein.
- Die Ressource, für die Sie einen Trace erstellen möchten, darf nicht das Kommunikationsverfahren über Signale verwenden.
- Die Ressource, für die Sie einen Trace erstellen möchten, muss so programmiert sein, dass sie auf die Traceanforderung reagiert.

Tasks für das Aktivieren/Inaktivieren eines Subsystems, einer Subsystemgruppe oder eines Subservers

<i>Task</i>	<i>SMIT-Direktaufruf</i>	<i>Befehl oder Datei</i>
Traceerstellung für ein Subsystem aktivieren (Kurzformat)	<b>smit tracesyson</b>	<b>traceson -s Subsystem</b>
Traceerstellung für ein Subsystem aktivieren (Langformat)	<b>smit tracesyson</b>	<b>traceson -l -s Subsystem</b>
Traceerstellung für ein Subsystem inaktivieren	<b>smit tracesysoff</b>	<b>tracesoff -s Subsystem</b>

### Zugehörige Tasks:

„System Resource Controller starten“ auf Seite 194

Der System Resource Controller (SRC) wird während der Systeminitialisierung über einen Datensatz für den Dämon `/usr/sbin/srcmstr` in der Datei `/etc/inittab` gestartet.

## Betriebssystemdateien

Dateien werden für die Eingabe und Ausgabe (E/A) aller Informationen im Betriebssystem verwendet, um den Zugriff auf Software und Hardware zu standardisieren.

Eine *Eingabe* erfolgt, wenn der Inhalt einer Datei geändert oder in die Datei geschrieben wird. Eine *Ausgabe* erfolgt, wenn der Inhalt einer Datei gelesen oder in eine andere Datei übertragen wird. Um beispielsweise ein Druckexemplar einer Datei zu erstellen, liest das System die Informationen aus der Textdatei und schreibt diese Informationen in die Druckdatei.

### Dateitypen

Die folgenden Typen von Dateien werden vom System erkannt: **reguläre Dateien**, **Verzeichnisdateien** und **Geräte-dateien**. Das Betriebssystem verwendet jedoch viele Varianten dieser grundlegenden Dateitypen.

Die folgenden Basisdateitypen sind verfügbar:

Eintrag	Beschreibung
<b>Reguläre Dateien</b>	Speichern Daten (Text-, binäre und ausführbare Daten).
<b>Verzeichnisdateien</b>	Enthalten Informationen zum Zugriff auf andere Dateien.
<b>Gerätedateien</b>	Definieren eine FIFO-Datei (First In/First Out) oder eine physische Einheit.

Alle vom System verwendeten Dateitypen gehören einer dieser Kategorien an. Das Betriebssystem verwendet jedoch viele Varianten dieser grundlegenden Dateitypen.

## Reguläre Dateien

Reguläre Dateien sind die am häufigsten vorkommenden Dateien und werden zum Speichern von Daten verwendet. Reguläre Dateien können Textdateien oder Binärdateien sein.

### Textdateien

Textdateien sind reguläre Dateien, die im ASCII-Format gespeicherte Daten enthalten und vom Benutzer gelesen werden können. Textdateien können angezeigt und gedruckt werden. Die Zeilen einer Textdatei dürfen keine NUL-Zeichen enthalten und die mit {LINE\_MAX} festgelegte Länge (Bytes), einschließlich des Zeilenvorschubzeichens nicht überschreiten.

Der Begriff *Textdatei* schließt nicht aus, dass in einer solchen Datei auch Steuerzeichen oder andere nicht druckbare Zeichen (außer NUL) enthalten sind. Sollten die Sonderzeichen von den Standarddienstprogrammen, die Textdateien als Eingabe oder Ausgabe auflisten, nicht verarbeitet werden können, sind die entsprechenden Einschränkungen in den Abschnitten zu diesen Dienstprogrammen beschrieben.

### Binärdateien

Binärdateien sind reguläre Dateien, die für den Computer lesbare Informationen enthalten. Es kann sich dabei um ausführbare Dateien handeln, mit denen das System angewiesen wird, einen Job auszuführen. Befehle und Programme werden in ausführbaren Binärdateien gespeichert. ASCII-Text wird von speziellen Compiler-Programmen in Binärcode umgesetzt.

Die einzigen Unterschiede zwischen Text- und Binärdateien sind die, dass die Zeilen in Textdateien bei der Länge auf den Wert (Bytes) von {LINE\_MAX} beschränkt sind, keine NUL-Zeichen enthalten dürfen und mit einem Zeilenvorschubzeichen abgeschlossen werden müssen.

## Verzeichnisdateien

Verzeichnisdateien enthalten Informationen, die das System benötigt, um auf alle Arten von Dateien zugreifen zu können. Sie enthalten nicht die eigentlichen Dateidaten. Deshalb belegen Verzeichnisse weniger Speicherplatz als reguläre Dateien und verleihen der Dateisystemstruktur mehr Flexibilität und Tiefe. Ein Verzeichniseintrag repräsentiert entweder eine Datei oder ein Unterverzeichnis. Jeder Eintrag enthält den Namen und die *I-Node*-Referenznummer der Datei. Die I-Node-Nummer zeigt auf den eindeutigen I-Node, der der Datei zugeordnet ist. Die I-Node-Nummer beschreibt die Position der Dateien, die der Datei zugeordnet sind. Für die Erstellung und Steuerung von Verzeichnissen wird eine eigene Befehlsgruppe verwendet.

## Gerätedateien

Gerätedateien definieren die Einheiten für das System oder die von Prozessen erstellten temporären Dateien. Die Basistypen für Gerätedateien sind FIFO-Dateien (First-in, First-out), blockorientierte Dateien und zeichenorientierte Dateien. FIFO-Dateien werden auch als *Pipes* (Befehlsverkettung) bezeichnet. Pipes werden von einem Prozess erstellt, um eine vorübergehende Kommunikation mit einem anderen Prozess zuzulassen. Diese Dateien gehen nach Beendigung des ersten Prozesses verloren. Block- und zeichenorientierte Dateien definieren Einheiten.

Jede Datei besitzt eine Gruppe von Berechtigungen (oder *Zugriffsmodi*), die bestimmen, wer die Datei lesen, ändern und ausführen darf.

### Zugehörige Konzepte:

„Zugriffsmodi für Dateien und Verzeichnisse“ auf Seite 314

Jede Datei hat einen Eigner. Wenn eine neue Datei erstellt wird, ist der Benutzer, der die Datei erstellt, der Eigner der Datei. Der Eigner legt einen *Zugriffsmodus* für die Datei fest. Mit den Zugriffsmodi können anderen Systembenutzern die Berechtigungen zum Lesen, Ändern oder Ausführen der Datei erteilt werden. Der Zugriffsmodus einer Datei kann nur vom Dateieigner oder von Benutzern mit Rootberechtigung geändert werden.

### Namenskonventionen für Dateien:

Jede Datei muss in dem Verzeichnis, in dem sie gespeichert wird, einen eindeutigen Namen besitzen. Auf diese Weise wird sichergestellt, dass die Datei einen eindeutigen Pfadnamen innerhalb des Dateisystems hat.

Die folgenden Richtlinien sind beim Benennen von Dateien zu beachten:

- Ein Dateiname darf aus maximal 255 Zeichen bestehen und kann Buchstaben, Zahlen und Unterstrichungszeichen enthalten.
- Das Betriebssystem unterscheidet grundsätzlich zwischen Groß- und Kleinschreibung. Deshalb sind auch FILEA, FiLea und filea drei unterschiedliche Dateinamen, wobei sich die Dateien durchaus in einem Verzeichnis befinden können.
- Dateinamen sollten so deutlich wie möglich auf den Inhalt der Datei hinweisen.
- Für Verzeichnisse gelten dieselben Namenskonventionen wie für Dateien.
- Manche Zeichen haben eine spezielle Bedeutung für das Betriebssystem. Vermeiden Sie es deshalb, diese Zeichen in Dateinamen zu verwenden. Dabei handelt es sich um folgende Zeichen:  
/ \ " ' \* ; - ? [ ] ( ) ~ ! \$ { } &lt; # @ & | Leerschritt Tabulator Zeilenumbruch
- Ein Dateiname ist bei der normalen Auflistung eines Verzeichnisses verdeckt, wenn er mit einem Punkt (.) beginnt. Wenn Sie den Befehl `ls` mit dem Flag `-a` eingeben, werden alle verdeckten Daten zusammen mit den regulären Dateien und Verzeichnissen aufgelistet.

### Dateipfadnamen:

Der Pfadname jeder Datei und jedes Verzeichnisses im Dateisystem setzt sich aus den Namen aller Verzeichnisse, die der Datei in der Baumstruktur übergeordnet sind, zusammen.

Da alle Pfade in einem Dateisystem vom Stammverzeichnis (/) ausgehen, besteht zwischen jeder Datei im Dateisystem und dem Stammverzeichnis eine eindeutige Anordnungsbeziehung, die als *absoluter Pfadname* bezeichnet wird. Absolute Pfadnamen beginnen mit einem Schrägstrich (/). Ein Beispiel für den absoluten Pfadnamen einer Datei `h` ist `/B/C/h`. Es können durchaus zwei Dateien mit dem Namen `h` im System existieren. Da die absoluten Pfade der beiden Dateien verschieden sind, z. B. `/B/h` und `/B/C/h`, hat jede Datei `h` einen eindeutigen Namen im System. Mit Ausnahme der letzten Komponente sind alle Komponenten eines Pfadnamens Verzeichnisse. Die letzte Komponente eines Pfadnamens kann ein Dateiname sein.

**Anmerkung:** Pfadnamen dürfen nicht länger als 1023 Zeichen sein.

### Mustererkennung mit Platzhalter- und Metazeichen:

Die Verwendung von Platzhalterzeichen vereinfacht die Angabe mehrerer Datei- oder Verzeichnisnamen.

Die gültigen Platzhalterzeichen sind der Stern (\*) und das Fragezeichen (?). Gültige Metazeichen sind linke und rechte eckige Klammern ([ ]), das Minuszeichen (-) und das Ausrufezeichen (!).

### *Mustererkennung mit dem Platzhalterzeichen \*:*

Mit dem Stern (\*) kann eine beliebige Zeichenfolge ersetzt werden.

Der Stern (\*) kann für jede Zeichenfolge einschließlich leerer Zeichenfolgen stehen.

Beispiele:

- Beispiel: Folgende Dateien befinden sich im Verzeichnis:

1test 2test aDatei1 aDatei2 bDatei1 Datei Datei1 Datei10 Datei2 Datei3

Wenn Sie nur die Dateien auswählen möchten, die mit Datei beginnen, geben Sie Folgendes ein:

Datei\*

Die folgenden Dateien werden ausgewählt: Datei, Datei1, Datei10, Datei2 und Datei3.

- Wenn Sie nur die Dateien auswählen möchten, die das Wort Datei enthalten, geben Sie Folgendes ein:

\*Datei\*

Die folgenden Dateien werden ausgewählt: aDatei1, aDatei2, bDatei1, Datei, Datei1, Datei10, Datei2 und Datei3.

### *Mustererkennung mit dem Platzhalterzeichen ?:*

Das Fragezeichen (?) kann ein beliebiges Einzelzeichen ersetzen.

Das Fragezeichen (?) kann ein beliebiges Einzelzeichen ersetzen. Beispiele:

- Wenn Sie nur die Dateien auswählen möchten, die mit **Datei** beginnen und danach noch ein weiteres Zeichen enthalten, geben Sie Folgendes ein:

Datei?

Die folgenden Dateien werden ausgewählt: Datei1, Datei2, Datei3.

- Wenn Sie nur die Dateien auswählen möchten, die mit **Datei** beginnen und danach noch zwei weitere Zeichen enthalten, geben Sie Folgendes ein:

Datei??

Die folgende Datei wird ausgewählt: Datei10.

### *Mustererkennung mit den Shellmetazeichen [ ]:*

Metazeichen sind ein weiterer Typ von Notationen mit Platzhalterzeichen, bei dem die gesuchten Zeichen in eckige Klammern ([ ]) eingeschlossen sind. Die Verwendung ist ähnlich wie beim Fragezeichen (?), allerdings können Sie in dieser Notation bestimmte zu suchende Zeichen angeben.

Sie können in den eckigen Klammern ([ ]) auch einen Wertebereich angeben. Hierfür verwenden Sie das Minuszeichen (-). Wenn Sie alle Buchstaben des Alphabets angeben möchten, verwenden Sie `[[:alpha:]]`. Wenn Sie alle Kleinbuchstaben des Alphabets angeben möchten, verwenden Sie `[[:lower:]]`.

Beispiele:

- Geben Sie Folgendes ein, um ausschließlich die Dateien auszuwählen, die mit 1 oder 2 enden:

\*Datei[12]

Die folgenden Dateien werden ausgewählt: aDatei1, aDatei2, Datei1 und Datei2.

- Geben Sie Folgendes ein, um ausschließlich die Dateien auszuwählen, die mit einer Ziffer beginnen:

[0123456789]\* **oder** [0-9]\*



Die folgenden Dateien werden ausgewählt: 1test und 2test.

- Geben Sie Folgendes ein, wenn Sie ausschließlich die Dateien auswählen möchten, die nicht mit einem a beginnen:

[!a]\*

Die folgenden Dateien werden ausgewählt: 1test, 2test, bDatei1, Datei, Datei1, Datei10, Datei2 und Datei3.

### Mustererkennung im Vergleich mit regulären Ausdrücken:

Mit regulären Ausdrücken können spezifische Zeichenfolgen aus einer Menge von Zeichenfolgen ausgewählt werden. Reguläre Ausdrücke werden in der Regel in der Textverarbeitung benutzt.

Sie können eine Vielzahl möglicher Zeichenfolgen darstellen. Da viele reguläre Ausdrücke je nach Locale unterschiedlich interpretiert werden können, werden so genannte Internationalisierungsfunktionen für die Kontextinvarianz innerhalb der Locales bereitgestellt.

Beispielvergleich:

Mustererkennung	Regulärer Ausdruck
*	.*
?	.
[!a]	[^a]
[abc]	[abc]
[:a\pha:]]	[[:a\pha:]]

Die vollständige Syntax des Befehls **awk** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

### Dateien verwalten

Es gibt zahlreiche Verwendungsmöglichkeiten für die Dateien in Ihrem System. Gewöhnlich wird eine Textdatei mit einem Texteditor erstellt.

Die bekannten Editoren in der UNIX-Umgebung sind vi und ed. Da verschiedene Texteditoren verfügbar sind, liegt die Entscheidung, welcher verwendet wird, bei Ihnen.

Sie können Dateien auch durch die Verwendung von Eingabe- und Ausgabeumleitung erstellen. Sie können die Ausgabe eines Befehls an eine neue Datei senden oder an eine vorhandene Datei anfügen.

Nach der Erstellung oder Änderung von Dateien müssen diese möglicherweise von einem Verzeichnis in ein anderes kopiert bzw. verschoben oder umbenannt werden. Möglicherweise müssen auch neue Verzeichnisse erstellt werden, wenn mit mehreren Projekten gearbeitet wird.

Bestimmte Dateien müssen ggf. gelöscht werden. Das Verzeichnis kann schnell mit Dateien übersät sein, die veraltete oder wertlose Informationen enthalten. Vergewissern Sie beim Freigeben von Speicherplatz, dass Sie nur Dateien löschen, die nicht mehr benötigt werden:

#### Zugehörige Konzepte:

„Eingabe- und Ausgabeumleitung“ auf Seite 362

Im Betriebssystem AIX können Sie die Eingabe und Ausgabe (E/A) von Daten mit bestimmten E/A-Befehlen und -Symbolen steuern.

#### Dateien löschen (Befehl rm):

Mit dem Befehl **rm** können Sie Dateien, die Sie nicht mehr benötigen, entfernen.

Der Befehl **rm** entfernt die Einträge für eine bestimmte Datei, Gruppe von Dateien oder bestimmte Dateien, die Sie aus einer Liste in einem Verzeichnis auswählen. Sie benötigen keine Benutzerbestätigung, Leseberechtigung und Schreibberechtigung, wenn Sie eine Datei mit dem Befehl **rm** entfernen möchten. Allerdings müssen Sie Schreibberechtigung für das Verzeichnis besitzen, in dem die Datei enthalten ist.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **rm**:

- Geben Sie Folgendes ein, um die Datei mit dem Namen `meinedatei` zu löschen:  
`rm minedatei`
- Geben Sie Folgendes ein, um nacheinander alle Dateien im Verzeichnis `meinverz` zu löschen:  
`rm -i meinverz/*`

Die Dateinamen werden nacheinander angezeigt. Geben Sie für jede Datei ein `j` ein, und drücken Sie anschließend die Eingabetaste, um die Datei zu löschen. Wenn Sie die Datei behalten möchten, drücken Sie nur die Eingabetaste.

Die vollständige Syntax des Befehls **rm** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

### **Dateien verschieben und umbenennen (Befehl mv):**

Mit dem Befehl **mv** können Sie Dateien und Verzeichnisse von einem Verzeichnis in ein anderes verschieben oder eine Datei oder ein Verzeichnis umbenennen. Wenn Sie eine Datei oder ein Verzeichnis in ein neues Verzeichnis verschieben, ohne einen neuen Namen anzugeben, wird der ursprüngliche Dateiname beibehalten.

**Achtung:** Der Befehl **mv** kann viele vorhandene Dateien überschreiben, wenn Sie das Flag **-i** nicht angeben. Das Flag **-i** fordert von Ihnen eine Bestätigung an, bevor eine Datei überschrieben wird. Wenn Sie das Flag **-f** angeben, wird keine Bestätigung von Ihnen angefordert. Wenn das Flag **-f** und das Flag **-i** angegeben werden, hat das zuletzt angegebene Flag Priorität.

#### **Dateien mit dem Befehl mv verschieben**

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **mv**:

- Wenn Sie beispielsweise eine Datei in ein anderes Verzeichnis verschieben und der Datei einen neuen Namen zuweisen möchten, geben Sie den folgenden Befehl ein:  
`mv intro manual/chap1`

Dieser Befehl verschiebt die Datei `intro` in das Verzeichnis `manual/chap1`. Der Name **intro** wird aus dem aktuellen Verzeichnis entfernt, und dieselbe Datei erscheint als `chap1` im Verzeichnis `manual`.

- Geben Sie Folgendes ein, um eine Datei in ein anderes Verzeichnis zu verschieben:  
`mv chap3 manual`

Dieser Befehl verschiebt `chap3` nach `manual/chap3`.

#### **Dateien mit dem Befehl mv umbenennen**

Mit dem Befehl **mv** können Sie den Namen einer Datei ändern, ohne sie in ein anderes Verzeichnis zu verschieben.

Geben Sie Folgendes ein, um eine Datei umzubenennen:

```
mv appendix apndx.a
```

Dieser Befehl benennt die Datei `appendix` in `apndx.a` um. Wenn bereits eine Datei mit dem Namen `apndx.a` vorhanden ist, wird der alte Inhalt dieser Datei durch den Inhalt der Datei `appendix` ersetzt.

Die vollständige Syntax des Befehls **mv** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

### Dateien kopieren (Befehl **cp**):

Mit dem Befehl **cp** können Sie den Inhalt der bzw. des mit dem Parameter *Quellendatei* bzw. *Quellenverzeichnis* angegebenen Datei bzw. Verzeichnisses in die bzw. das mit dem Parameter *Zieldatei* bzw. *Zielverzeichnis* angegebene Datei bzw. Verzeichnis kopieren.

Falls die als *Zieldatei* angegebene Datei vorhanden ist, wird die Kopie ohne Warnung über den Originalinhalt der Datei geschrieben. Wenn Sie mehrere *Quellendateien* kopieren, muss das Ziel ein Verzeichnis sein.

Falls im Zielverzeichnis bereits eine Datei mit diesem Namen vorhanden ist, überschreibt die kopierte Datei die Datei im Zielverzeichnis. Deshalb empfiehlt es sich, der Dateikopie einen *neuen* Namen zu geben, um sicherzustellen, dass im Zielverzeichnis keine Datei mit demselben Namen vorhanden ist.

Wenn Sie die *Quellendatei* in ein Verzeichnis kopieren möchten, müssen Sie den Pfad eines vorhandenen *Zielverzeichnisses* angeben. Dateien behalten ihren Namen, wenn sie in ein anderes Verzeichnis kopiert werden, sofern am Ende des Pfads kein neuer Dateiname angegeben wird. Der Befehl **cp** kopiert auch vollständige Verzeichnisse in andere Verzeichnisse, wenn Sie das Flag **-r** oder **-R** angeben.

Mit dem Flag **-R** können Sie auch Gerätedateien kopieren. Wenn Sie das Flag **-R** angeben, werden die Gerätedateien unter dem neuen Pfadnamen neu erstellt. Der Befehl **cp** versucht, die Gerätedateien in reguläre Dateien zu kopieren, wenn Sie das Flag **-r** angeben.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **cp**:

- Geben Sie Folgendes ein, um eine Kopie einer Datei im aktuellen Verzeichnis zu erstellen:

```
cp prog.c prog.bak
```

Dieser Befehl kopiert die Datei `prog.c` in die Datei `prog.bak`. Wenn die Datei `prog.bak` noch nicht vorhanden ist, wird sie vom Befehl **cp** erstellt. Ist die Datei vorhanden, ersetzt der Befehl **cp** sie durch eine Kopie der Datei `prog.c`.

- Geben Sie Folgendes ein, um eine Datei aus dem aktuellen Verzeichnis in ein anderes Verzeichnis zu kopieren:

```
cp jones /home/nick/clients
```

Dieser Befehl kopiert die Datei `jones` in das Verzeichnis `/home/nick/clients/jones`.

- Geben Sie Folgendes ein, um alle Dateien aus einem Verzeichnis in ein neues Verzeichnis zu kopieren:

```
cp /home/janet/clients/* /home/nick/customers
```

Dieser Befehl kopiert nur die Dateien im Verzeichnis `clients` in das Verzeichnis `customers`.

- Geben Sie Folgendes ein, um eine bestimmte Gruppe von Dateien in ein anderes Verzeichnis zu kopieren:

```
cp jones lewis smith /home/nick/clients
```

Dieser Befehl kopiert die Dateien `jones`, `lewis` und `smith` aus dem aktuellen Arbeitsverzeichnis in das Verzeichnis `/home/nick/clients`.

- Wenn Sie für das Kopieren von Dateien Platzhalterzeichen verwenden möchten, geben Sie Folgendes:

```
cp programs/*.c .
```

Dieser Befehl kopiert die Dateien aus dem Verzeichnis `programs`, die mit `.c` enden, in das aktuelle Verzeichnis (das durch den Punkt `.` dargestellt ist). Zwischen `c` und dem abschließenden Punkt muss ein Leerzeichen eingegeben werden.

Die vollständige Syntax des Befehls **cp** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

### Dateien suchen (Befehl **find**):

Mit dem Befehl **find** können Sie in der Verzeichnisstruktur rekursiv in einem angegebenen *Pfad* nach Dateien suchen, die einem booleschen Ausdruck entsprechen, der den im Folgenden beschriebenen Bedingungen entspricht.

Die Ausgabe des Befehls **find** richtet sich nach den im *Ausdruck* angegebenen Bedingungen.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **find**:

- Wenn Sie alle Dateien im Dateisystem auflisten möchten, die den Namen `.profile` haben, geben Sie Folgendes ein:

```
find / -name .profile
```

Dieser Befehl durchsucht das gesamte Dateisystem und gibt die vollständigen Pfadnamen aller Dateien mit dem Namen `.profile` aus. Der Schrägstrich (/) weist den Befehl **find** an, das Stammverzeichnis (/) und alle Unterverzeichnisse zu durchsuchen.

Um Zeit einzusparen, können Sie die Suche durch Angabe der Verzeichnisse, in denen sich die Dateien Ihrer Meinung nach befinden müssten, einschränken.

- Wenn Sie die Dateien in der aktuellen Verzeichnisstruktur auflisten möchten, die den speziellen Berechtigungscode `0600` haben, geben Sie Folgendes ein:

```
find . -perm 0600
```

Dieser Befehl listet die Namen der Dateien auf, für die *nur* der Eigner lese- und schreibberechtigt ist. Der Punkt (.) weist den Befehl **find** an, das aktuelle Verzeichnis und dessen Unterverzeichnisse zu durchsuchen. Eine Erläuterung der Berechtigungs\_codes finden Sie in der Beschreibung des Befehls **chmod**.

- Wenn Sie mehrere Verzeichnisse nach Dateien mit bestimmten Berechtigungs\_codes durchsuchen möchten, geben Sie Folgendes ein:

```
find manual clients proposals -perm -0600
```

Dieser Befehl listet die Namen der Dateien auf, die Lese- und Schreibberechtigung für den Eigner und möglicherweise weitere Berechtigungen besitzen. Es werden die Verzeichnisse `manual`, `clients` und `proposals` sowie die zugehörigen Unterverzeichnisse durchsucht. Im vorherigen Beispiel wählt `-perm 0600` nur die Dateien aus, die exakt den Berechtigungscode `0600` haben. In diesem Beispiel wählt `-perm -0600` Dateien mit Berechtigungs\_codes aus, die Zugriffe, die dem Code `0600` entsprechen, und weitere Zugriffe zulassen, die über die Berechtigungsstufe `0600` hinausgehen. Der Ausdruck gilt also auch für die Berechtigungs\_codes `0622` und `2744`.

- Wenn Sie alle Dateien im aktuellen Verzeichnis auflisten möchten, die während der letzten 24 Stunden geändert wurden, geben Sie Folgendes ein:

```
find . -ctime 1
```

- Wenn Sie reguläre Dateien mit mehreren Verbindungen suchen möchten, geben Sie Folgendes ein:

```
find . -type f -links +1
```

Dieser Befehl listet die Namen der gewöhnlichen Dateien (`-type f`) auf, die mehr als eine Verbindung haben (`-links +1`).

**Anmerkung:** Jedes Verzeichnis hat mindestens zwei Verbindungen, den Eintrag im Elternverzeichnis und seinen eigenen Eintrag (.). Weitere Informationen zu mehreren Dateiverbindungen finden Sie in der Beschreibung des Befehls **ln**.

- Wenn Sie alle Dateien mit einer Größe von exakt 414 Bytes suchen möchten, geben Sie Folgendes ein:

```
find . -size 414c
```

Die vollständige Syntax des Befehls **find** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

### **Dateityp anzeigen (Befehl file):**

Mit dem Befehl **file** können Sie die mit dem Parameter *Datei* oder *-fDateiliste* angegebenen Dateien lesen, eine Reihe von Dateitests durchführen und versuchen, die Dateien nach Typ zu klassifizieren. Anschließend schreibt der Befehl die Dateitypen in die Standardausgabe.

Wenn es sich bei der Datei um eine ASCII-Datei zu handeln scheint, überprüft der Befehl **file** die ersten 512 Bytes und bestimmt die Sprache. Scheint es sich nicht um eine ASCII-Datei zu handeln, versucht der Befehl **file** festzustellen, ob die Datei eine Binärdatei oder eine Textdatei mit erweiterten Zeichen ist.

Wenn der Parameter *Datei* eine ausführbare Datei oder Objektmoduldatei mit einer Versionsnummer größer als 0 angibt, zeigt der Befehl **file** die Versionsnummer an.

Der Befehl **file** verwendet die Datei */etc/magic* für die Identifizierung von Dateien, die einen Dateitypanzeiger haben, d. h. alle Dateien, die eine numerische oder Zeichenfolgekonstante enthalten, die den Typ angibt.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **file**:

- Geben Sie Folgendes ein, um die Typinformationen der Datei mit dem Namen *meinedatei* anzuzeigen:

```
file meineDatei
```

Dieser Befehl zeigt den Dateityp von *meinedatei* an (z. B. Verzeichnis, Daten, ASCII-Text, C-Programmquelle oder Archiv).

- Geben Sie Folgendes ein, um den Typ der in der Datei *Dateinamen.lst* angegebenen Dateien anzuzeigen:

```
file -f Dateinamen.lst
```

Dieser Befehl zeigt den Typ aller Dateien an, die in der Datei *Dateinamen.lst* aufgeführt sind. Jeder Dateiname muss in einer separaten Zeile angezeigt werden.

- Wenn Sie eine Datei *Dateinamen.lst* erstellen möchten, die die Namen aller Dateien im aktuellen Verzeichnis enthält, geben Sie Folgendes ein:

```
ls > Dateinamen.lst
```

Sie können die Datei *Dateinamen.lst* wie gewünscht ändern.

Die vollständige Syntax des Befehls **file** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

### **Befehle für das Anzeigen von Dateiinhalten (Befehle pg, more, page und cat):**

Mit den Befehlen **pg**, **more** und **page** können Sie den Inhalt einer Datei anzeigen und die Geschwindigkeit steuern, mit der die Dateien angezeigt werden.

Außerdem können Sie mit dem Befehl **cat** den Inhalt einer oder mehrerer Dateien am Bildschirm anzeigen. Wenn Sie den Befehl **cat** mit dem Befehl **pg** kombinieren, können Sie den Inhalt einer Datei seitenweise anzeigen.

Sie können den Inhalt von Dateien auch durch Eingabe-/Ausgabeumleitung anzeigen.

### **Zugehörige Konzepte:**

„Eingabe- und Ausgabeumleitung“ auf Seite 362

Im Betriebssystem AIX können Sie die Eingabe und Ausgabe (E/A) von Daten mit bestimmten E/A-Befehlen und -Symbolen steuern.

*Befehl pg verwenden:*

Der Befehl **pg** liest die Dateinamen aus dem Parameter **Datei** und schreibt sie seitenweise in die Standardausgabe.

Wenn Sie das Minuszeichen (-) für den Parameter **Datei** angeben oder den Befehl **pg** ohne Optionen ausführen, liest der Befehl **pg** die Standardeingabe. Am Ende jeder angezeigten Seite steht eine Eingabeaufforderung. Wenn Sie die Eingabetaste drücken, wird eine weitere Seite angezeigt. Mit den Unterbefehlen für den Befehl **pg** können Sie bereits angezeigte Teile des Inhalts erneut anzeigen.

Geben Sie Folgendes ein, um den Inhalt der Datei `meinedatei` seitenweise anzuzeigen:

```
pg minedatei
```

Die vollständige Syntax des Befehls **pg** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

*Befehle more und page verwenden:*

Die Befehle **more** und **page** zeigen fortlaufenden Text seitenweise an.

Sie halten den Anzeigevorgang nach jeder Seite an und geben unten auf der Seite den *Dateinamen* sowie den bereits angezeigten Inhalt der Datei in Prozent aus (z. B. `meine Datei (7%)`). Wenn Sie die Eingabetaste drücken, zeigt der Befehl **more** eine weitere Zeile an. Wenn Sie die Leertaste drücken, zeigt der Befehl **more** eine weitere Seite mit Text an.

**Anmerkung:** Bei einigen Terminalmodellen löscht der Befehl **more** vor der Anzeige der nächsten Textseite den Inhalt der Anzeige, anstatt zu blättern.

Wenn Sie beispielsweise eine Datei mit dem Namen `meinedatei` anzeigen möchten, geben Sie Folgendes ein:

```
more minedatei
```

Drücken Sie die Leertaste, um die nächste Seite anzuzeigen.

Die vollständige Syntax des Befehls **more** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

*Befehl cat verwenden:*

Der Befehl **cat** liest die mit dem Parameter *Datei* angegebenen Dateien der Reihe nach und schreibt sie in die Standardausgabe.

Beispiele:

- Geben Sie Folgendes ein, um den Inhalt der Datei `notes` anzuzeigen:

```
cat notes
```

Enthält die Datei mehr als 24 Zeilen, verschieben sich einige Zeilen aus dem Anzeigebereich. Wenn Sie eine Datei seitenweise anzeigen möchten, verwenden Sie den Befehl **pg**.

- Wenn Sie den Inhalt der Dateien `notes`, `notes2` und `notes3` anzeigen möchten, geben Sie Folgendes ein:

```
cat notes notes2 notes3
```

Die vollständige Syntax des Befehls **cat** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

### Textzeichenfolgen in Dateien suchen (Befehl **grep**):

Mit dem Befehl **grep** können Sie die angegebene Datei nach einem bestimmten *Muster* durchsuchen. Der Befehl schreibt alle übereinstimmenden Zeilen in die Standardausgabe.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **grep**:

- Wenn Sie in einer Datei mit dem Namen `pgm.s` nach einem Muster suchen möchten, das Platzhalterzeichen (`*`, `^`, `?`, `[`, `]`, `\(`, `\)`, `\{` oder `\}`) enthält, und wie in diesem Fall Zeilen, die mit einem Groß- oder Kleinbuchstaben beginnen, geben Sie Folgendes ein:

```
grep "^[a-zA-Z]" pgm.s
```

Dieser Befehl zeigt alle Zeilen in der Datei `pgm.s` an, die mit einem Buchstaben beginnen.

- Wenn Sie alle Zeilen in der Datei `sort.c` anzeigen möchten, die einem bestimmten Muster nicht entsprechen, geben Sie Folgendes ein:

```
grep -v bubble sort.c
```

Dieser Befehl zeigt alle Zeilen in der Datei `sort.c` an, die das Wort `bubble` nicht enthalten.

- Wenn Sie alle Zeilen in der Ausgabe des Befehls **ls** anzeigen möchten, die der Zeichenfolge `staff` entsprechen, geben Sie Folgendes ein:

```
ls -l | grep staff
```

Die vollständige Syntax des Befehls **grep** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

### Textdateien sortieren (Befehl **sort**):

Mit dem Befehl **sort** können Sie Zeilen in Dateien (die Sie mit dem Befehl angeben) alphabetisch sortieren und das Ergebnis in die Standardausgabe schreiben.

Wenn Sie mit dem Befehl mehrere **Dateien** angeben, verknüpft der Befehl **sort** die Dateien und sortiert die Zeilen in den Dateien alphabetisch so, als gehörten sie zu einer Datei.

**Anmerkung:** Der Befehl **sort** berücksichtigt die Klein- und Großschreibung und ordnet Großbuchstaben vor Kleinbuchstaben ein (abhängig von der Locale).

In den folgenden Beispielen hat die Datei `names` den folgenden Inhalt:

```
marta
denise
joyce
endrica
melanie
```

Die Datei `states` hat den folgenden Inhalt:

```
texas
colorado
ohio
```

- Geben Sie Folgendes ein, um den sortierten Inhalt der Datei `names` anzuzeigen:

```
sort names
```

Das System zeigt Informationen ähnlich den folgenden an:

```
denise
endrica
joyce
marta
melanie
```

- Geben Sie Folgendes ein, um den sortierten Inhalt der Dateien `names` und `states` anzuzeigen:  
`sort names states`

Das System zeigt Informationen ähnlich den folgenden an:

```
colorado
denise
endrica
joyce
marta
melanie
ohio
texas
```

- Geben Sie Folgendes ein, um den ursprünglichen Inhalt der Datei `names` durch den sortierten Inhalt zu ersetzen:  
`sort -o names names`

Dieser Befehl ersetzt den Inhalt der Datei `names` durch den sortierten Inhalt.

Die vollständige Syntax des Befehls **sort** ist in der Veröffentlichung *Commands Reference, Volume 5* beschrieben.

### Dateien vergleichen (Befehl **diff**):

Mit dem Befehl **diff** können Sie Textdateien vergleichen. Es können sowohl Einzeldateien als auch gesamte Verzeichnisse verglichen werden.

Wenn Sie den Befehl **diff** für reguläre Dateien ausführen und Textdateien in unterschiedlichen Verzeichnissen miteinander vergleichen, teilt Ihnen der Befehl **diff** mit, welche Zeilen Sie in den Dateien ändern müssen, damit sie übereinstimmen.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **diff**:

- Geben Sie Folgendes ein, um zwei Dateien zu vergleichen:  
`diff chap1.bak chap1`

Dieser Befehl zeigt die Unterschiede zwischen den Dateien `chap1.bak` und `chap1` an.

- Geben Sie Folgendes ein, um zwei Dateien ohne Berücksichtigung von Leerzeichen zu vergleichen:  
`diff -w prog.c.bak prog.c`

Falls sich die Dateien nur in der Anzahl der Leer- und Tabulatorzeichen zwischen Wörtern unterscheiden, betrachtet der Befehl **diff -w** die Dateien als identisch.

Die vollständige Syntax des Befehls **diff** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

### Wörter, Zeilen und Bytes in Dateien zählen (Befehl **wc**):

Verwenden Sie den Befehl **wc**, um die Anzahl der Zeilen, Wörter und Bytes in den Dateien zu zählen, die mit dem Parameter *Datei* angegeben wurden.

Wenn Sie keine *Datei* angeben, wird die Standardeingabe verwendet. Der Befehl schreibt die Ergebnisse in die Standardausgabe und protokolliert die Gesamtsumme für alle angegebenen Dateien. Werden Flags



angegeben, bestimmt die Reihenfolge der Flags die Reihenfolge der Ausgabe. Ein *Wort* ist als eine Zeichenfolge definiert, die von Leerzeichen, Tabulatorzeichen oder Zeilenvorschubzeichen begrenzt wird.

Wenn in der Befehlszeile Dateien angegeben werden, werden die Namen dieser Dateien zusammen mit den Ergebnissen ausgegeben.

Beispiele:

- Geben Sie Folgendes ein, um die Anzahl der Zeilen, Wörter und Bytes der Datei chap1 anzuzeigen:

```
wc chap1
```

Dieser Befehl zeigt die Anzahl der Zeilen, Wörter und Bytes der Datei chap1 an.

- Geben Sie Folgendes ein, um nur die Anzahl der Wörter und Bytes anzuzeigen:

```
wc -cw chap*
```

Dieser Befehl zeigt die Anzahl der Bytes und Wörter jeder Datei an, deren Name mit chap beginnt. Die Gesamtsummen werden ebenfalls angezeigt.

Die vollständige Syntax des Befehls **wc** ist in der Veröffentlichung *Commands Reference, Volume 6* beschrieben.

### **Erste Zeilen von Dateien anzeigen (Befehl head):**

Der Befehl **head** schreibt die Anfangszeilen oder Bytes der angegebenen Dateien oder der Standardeingabe in die Standardausgabe.

Wenn Sie den Befehl **head** ohne Flags ausführen, werden standardmäßig die ersten 10 Zeilen angezeigt.

Geben Sie Folgendes ein, um die ersten fünf Zeilen der Datei Test anzuzeigen:

```
head -5 Test
```

Die vollständige Syntax des Befehls **head** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

### **Letzte Zeilen einer Datei anzeigen (Befehl tail):**

Verwenden Sie den Befehl **tail**, um die für den Parameter *Datei* angegebene Datei ab einer bestimmten Stelle in die Standardausgabe zu schreiben.

Beispiele:

- Geben Sie Folgendes ein, um die letzten 10 Zeilen der Datei notes anzuzeigen:

```
tail notes
```

- Geben Sie beispielsweise Folgendes ein, um die letzten 20 Zeilen der Datei notes anzuzeigen:

```
tail -20 notes
```

- Geben Sie Folgendes ein, um die Datei notes ab Byte 200 seitenweise anzuzeigen:

```
tail -c +200 notes | pg
```

- Geben Sie Folgendes ein, um das Wachstum der Datei accounts zu verfolgen:

```
tail -f accounts
```

Dieser Befehl zeigt die letzten 10 Zeilen der Datei accounts an. Sobald der Datei accounts Zeilen hinzugefügt werden, zeigt der Befehl **tail** diese an. Der Befehl wird so lange ausgeführt, bis die Tastenkombination Strg-C gedrückt wird.

Die vollständige Syntax des Befehls **tail** ist in der Veröffentlichung *Commands Reference, Volume 5* beschrieben.

### Abschnitte aus Textdateien ausschneiden (Befehl cut):

Mit dem Befehl **cut** können Sie ausgewählte Bytes, Zeichen oder Felder aus jeder Zeile einer Datei in die Standardausgabe schreiben.

Beispiele:

- Geben Sie beispielsweise den folgenden Befehl ein, um mehrere Felder aus jeder Zeile einer Datei anzuzeigen:

```
cut -f1,5 -d: /etc/passwd
```

Dieser Befehl zeigt die Felder für den Anmeldenamen und den vollständigen Benutzernamen aus der Datei mit den Systemkennwörtern an. Es handelt sich dabei um das erste und das fünfte Feld (-f1,5), die jeweils durch Doppelpunkte voneinander getrennt sind (-d:).

- Beispiel: Die Datei /etc/passwd enthält folgende Daten:

```
su:*:0:0:User with special privileges:/:usr/bin/sh
daemon:*:1:1::/etc:
bin:*:2:2::/usr/bin:
sys:*:3:3::/usr/src:
adm:*:4:4:system administrator:/var/adm:/usr/bin/sh
pierre:*:200:200:Pierre Harper:/home/pierre:/usr/bin/sh
joan:*:202:200:Joan Brown:/home/joan:/usr/bin/sh
```

Der Befehl **cut** bewirkt Folgendes:

```
su:User with special privileges
daemon:
bin:
sys:
adm:system administrator
pierre:Pierre Harper
joan:Joan Brown
```

Die vollständige Syntax des Befehls **cut** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

### Abschnitte aus Textdateien einfügen (Befehl paste):

Mit dem Befehl **paste** können Sie Zeilen aus bis zu 12 Dateien zu einer Datei zusammenfügen.

Beispiele:

- Beispiel: Die Datei names enthält folgenden Text:

```
rachel
jerry
mark
marsha
scott
```

Die Datei places enthält folgenden Text:

```
New York
Austin
Chicago
Boca Raton
Boca Raton
Seattle
```

Die Datei dates enthält folgenden Text:

February 5  
March 13  
June 21  
July 16  
November 4

Geben Sie Folgendes ein, um die Texte der Dateien `names`, `places` und `dates` in eine Datei einzufügen:  
`paste names places dates > npd`

Dieser Befehl erstellt eine Datei mit dem Namen `npd`, die die Daten aus der Datei `names` in einer Spalte, die Daten aus der Datei `places` in einer weiteren Spalte und die Daten aus der Datei `dates` in einer dritten Spalte enthält. Die Datei `npd` hat jetzt folgenden Inhalt:

```
rachel New York February 5
jerry Austin March 13
mark Chicago June 21
linda Boca Raton July 16
scott Seattle November 4
```

Name, Ort und Monat sind in einer Zeile jeweils durch ein Tabulatorzeichen voneinander getrennt. Die Spalten sind nicht ausgerichtet, da die Tabulatorzeichen an jeder achten Spalte gesetzt sind.

- Geben Sie Folgendes ein, um die Spalten durch ein anderes Zeichen als das Tabulatorzeichen zu trennen:

```
paste -d"!@" names places dates > npd
```

Dieser Befehl bewirkt, dass die Zeichen `!` und `@` abwechselnd als Trennzeichen zwischen den Spalten verwendet werden. Wenn beispielsweise die Dateien `names`, `places` und `dates` dieselben sind wie in Beispiel 1, wird der Inhalt der Datei `npd` wie folgt dargestellt:

```
rachel!New York@February 5
jerry!Austin@March 13
mark!Chicago@June 21
linda!Boca Raton@July 16
scott!Seattle@November 4
```

- Geben Sie Folgendes ein, um das aktuelle Verzeichnis vierspaltig anzuzeigen:

```
ls | paste - - - -
```

Für jedes Minuszeichen (`-`) erstellt der Befehl **paste** eine Spalte mit Daten, die aus der Standardeingabe gelesen werden. Die erste Zeile wird in die erste Spalte geschrieben, die zweite Zeile in die zweite Spalte usw.

Die vollständige Syntax des Befehls **paste** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

### **Zeilen in Textdateien nummerieren (Befehl nl):**

Der Befehl **nl** liest die angegebene Datei (standardmäßig die Standardeingabe), nummeriert die Zeilen in der Eingabe und schreibt die nummerierten Zeilen in die Standardausgabe.

Beispiele:

- Geben Sie Folgendes ein, um nur die Zeilen zu nummerieren, die nicht leer sind:

```
nl chap1
```

Dieser Befehl zeigt eine nummerierte Liste von `chap1` an, in der nur die nicht leeren Zeilen in den Hauptabschnitten nummeriert sind.

- Geben Sie beispielsweise Folgendes ein, um alle Zeilen zu nummerieren:

```
nl -ba chap1
```

Dieser Befehl nummeriert alle Zeilen in der Datei chap1 einschließlich der leeren Zeilen.

Die vollständige Syntax des Befehls `nl` ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

### Spalten in Textdateien entfernen (Befehl `colrm`):

Mit dem Befehl `colrm` können Sie Spalten aus einer Datei entfernen. Die Eingabe wird der Standardeingabe entnommen. Die Ausgabe wird an die Standardausgabe gesendet.

Wenn der Befehl mit einem Parameter aufgerufen wird, werden die Spalten ab der angegebenen Spalte bis zur letzten Spalte aus den Zeilen entfernt. Wenn der Befehl mit zwei Parametern aufgerufen wird, werden die Spalten ab der ersten angegebenen Spalte bis zur zweiten angegebenen Spalte gelöscht.

**Anmerkung:** Die Spaltennummerierung beginnt bei Spalte 1.

Beispiele:

- Geben Sie Folgendes ein, um Spalten aus der Datei `text.fil` zu löschen:

```
colrm 6 < text.fil
```

Die Datei `text.fil` enthält folgende Daten:

```
123456789
```

Der Befehl `colrm` zeigt Folgendes an:

```
12345
```

Die vollständige Syntax des Befehls `colrm` ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

## Datei- und Verzeichnisverbindungen

*Verbindungen* (Links) sind Verknüpfungen zwischen einem Dateinamen und einer I-Node-Referenznummer (I-Node-Nummer), der internen Darstellung einer Datei. Da Verzeichniseinträge Dateinamen mit I-Node-Nummern enthalten, stellt jeder Verzeichniseintrag eine Verbindung dar.

Dateien werden anhand der I-Node-Nummer und anhand des Dateinamens identifiziert. Durch die Verwendung von Verbindungen kann jede Datei bzw. jede I-Node-Nummer viele verschiedene Namen besitzen. Die I-Node-Nummer 798 beispielsweise enthält eine Kurzinformation bezüglich des Juni-Verkaufs im Büro Omaha. Derzeit lautet der Verzeichniseintrag für diese Kurzinformation wie folgt:

I-Node-Nummer	Dateiname
798	memo

Da diese Informationen zu Informationen gehören, die in den Verzeichnissen `verkauf` und `omaha` gespeichert sind, werden Verbindungen verwendet, um die Informationen an den Stellen zur Verfügung zu stellen, an denen Sie benötigt werden. Mit dem Befehl `ln` werden Verbindungen zu diesen Verzeichnissen erstellt. Die Datei besitzt nun drei Dateinamen:

I-Node-Nummer	Dateiname
798	memo
798	Verkauf/Juni
798	omaha/Juniverkauf

Wenn Sie mit dem Befehl **pg** oder **cat** den Inhalt der drei Dateien anzeigen, werden dieselben Informationen angezeigt. Wird der Inhalt des I-Nodes unter einem dieser drei Dateinamen geändert, gelten diese Änderungen auch für die anderen Dateinamen.

### Verbindungstypen:

Es gibt zwei Typen von Verbindungen: feste Verbindungen und symbolische Verbindungen.

Verbindungen werden mit dem Befehl **In** erstellt.

Eintrag	Beschreibung
<b>Feste Verbindung</b>	Eine feste Verbindung (oder Hard Link) ermöglicht den Zugriff auf Daten einer Datei über einen neuen Dateinamen. Feste Verbindungen stellen die Existenz von Dateien sicher. Wenn die letzte feste Verbindung entfernt wird, werden die I-Node-Nummer und die zugehörigen Daten gelöscht. Feste Verbindungen können nur zwischen Dateien erstellt werden, die sich im selben Dateisystem befinden.
<b>Symbolische Verbindung</b>	Ermöglicht den Zugriff auf Daten in anderen Dateisystemen mit einem neuen Dateinamen. Die symbolische Verbindung ist ein spezieller Dateityp, der einen Pfadnamen enthält. Wenn ein Prozess eine symbolische Verbindung findet, kann der Prozess diesen Pfad durchsuchen. Symbolische Verbindungen schützen eine Datei nicht davor, aus dem Dateisystem gelöscht zu werden.

**Anmerkung:** Der Benutzer, der eine Datei erstellt, bleibt Eigner der Datei, ungeachtet der Anzahl der erstellten Verbindungen. Der Zugriffsmodus für die Datei kann nur vom Eigner der Datei oder vom Rootbenutzer festgelegt werden. Über eine verknüpfte Datei können jedoch mit dem entsprechenden Zugriffsmodus Änderungen in der Datei vorgenommen werden.

Eine Datei bzw. ein Verzeichnis existiert so lange, wie eine feste Verbindung zur I-Node-Nummer dieser Datei besteht. In der mit dem Befehl **ls -l** angezeigten langen Liste wird die Anzahl der festen Verbindungen für jede Datei und jedes Unterverzeichnis angegeben. Alle festen Verbindungen werden vom Betriebssystem gleich behandelt, unabhängig vom Zeitpunkt der Erstellung.

### Dateien verbinden (Befehl **In**):

Das Verbinden von Dateien mit dem Befehl **In** empfiehlt sich, wenn Sie an mehreren Orten mit denselben Dateien arbeiten möchten.

Verbindungen werden erstellt, indem alternative Namen für die Originaldatei vergeben werden. Die Verwendung von Verbindungen ermöglicht die gemeinsame Verwendung von großen Dateien, wie z. B. einer Datenbank oder Mail-Liste, durch mehrere Benutzer, ohne Kopien der Datei erstellen zu müssen. Verbindungen sparen Plattenspeicherplatz ein. Außerdem werden die Änderungen in einer Datei automatisch in allen verbundenen Dateien nachvollzogen.

Mit dem Befehl **In** wird die im Parameter **Quelldatei** angegebene Datei mit der im Parameter **Zieldatei** angegebenen Datei oder mit derselben Datei in einem anderen Verzeichnis, das im Parameter **Zielverzeichnis** angegeben wird, verbunden. Standardmäßig erstellt der Befehl **In** feste Verbindungen. Wenn Sie den Befehl **In** zum Erstellen symbolischer Verbindungen verwenden möchten, fügen Sie das Flag **-s** hinzu.

**Anmerkung:** Dateien in verschiedenen Dateisystemen können ohne das Flag **-s** nicht verbunden werden.

Wird eine Datei mit einem neuen Namen verbunden, kann nur eine Datei angegeben werden. Wenn die Datei mit einem anderen Verzeichnis verbunden wird, können mehrere Dateien angegeben werden.

Der Parameter **Zieldatei** ist optional. Wenn Sie keine Zieldatei angeben, erstellt der Befehl **In** eine Datei im aktuellen Verzeichnis. Die neue Datei übernimmt den Namen der angegebenen **Quelldatei**.

Beispiele:

- Geben Sie Folgendes ein, um eine Verbindung zur Datei `chap1` zu erstellen:

```
In -f chap1 intro
```

Dieser Befehl verbindet die Datei `chap1` mit dem neuen Namen `intro`. Wenn Sie das Flag `-f` verwenden, wird eine Datei mit dem Namen `intro` erstellt, sofern diese noch nicht vorhanden ist. Falls die Datei `intro` vorhanden ist, wird die Datei durch eine Verbindung zu `chap1` ersetzt. Beide Dateinamen, `chap1` und `intro`, beziehen sich auf dieselbe Datei.

- Geben Sie Folgendes ein, um eine Datei mit dem Namen `index` mit einer Datei gleichen Namens im Verzeichnis `manual` zu verbinden:

```
In index manual
```

Dieser Befehl verbindet die Datei `index` mit dem neuen Namen `manual/index`.

- Geben Sie Folgendes ein, um mehrere Dateien in einem anderen Verzeichnis zu verbinden:

```
In chap2 jim/chap3 /home/manual
```

Dieser Befehl verbindet `chap2` mit dem neuen Namen `/home/manual/chap2` und `jim/chap3` mit dem Namen `/home/manual/chap3`.

- Geben Sie Folgendes ein, um den Befehl **In** mit Platzhalterzeichen zu verwenden:

```
In manual/* .
```

**Anmerkung:** Zwischen dem Stern und dem Punkt muss ein Leerzeichen eingefügt werden.

Dieser Befehl verbindet alle Dateien im Verzeichnis `manual` mit dem aktuellen Verzeichnis `.` (Punkt) und gibt den Dateien dort dieselben Namen wie im Verzeichnis `manual`.

- Geben Sie Folgendes ein, um eine symbolische Verbindung zu erstellen:

```
In -s /tmp/toc toc
```

Dieser Befehl erstellt im aktuellen Verzeichnis die symbolische Verbindung `toc`. Die Datei `toc` zeigt auf die Datei `/tmp/toc`. Wenn die Datei `/tmp/toc` vorhanden ist, listet der Befehl `cat toc` ihren Inhalt auf.

- Geben Sie Folgendes ein, um ohne Angabe des Parameters **Zieldatei** das gleiche Ergebnis zu erzielen:

```
In -s /tmp/toc
```

Die vollständige Syntax des Befehls **In** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

### **Befehl für das Entfernen von verknüpften Dateien:**

Der Befehl **rm** entfernt die angegebene verknüpfte Datei.

Wenn einer von mehreren Dateinamen mit einer festen Verbindung gelöscht wird, ist dies nicht gleichbedeutend mit dem Löschen der gesamten Datei, da diese weiter unter dem anderen Namen existiert. Nur wenn die letzte Verbindung zu einer I-Node-Nummer entfernt wird, werden auch die Daten entfernt. Die I-Node-Nummer ist damit zur erneuten Verwendung im System wieder verfügbar.

Die vollständige Syntax des Befehls **rm** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

### **DOS-Dateien**

Das Betriebssystem AIX unterstützt die Verwendung von DOS-Dateien auf dem System.

Kopieren Sie die gewünschten DOS-Dateien auf eine Diskette. Das System kann diese Dateien im richtigen Format in ein Verzeichnis des Basisbetriebssystems einlesen und anschließend im DOS-Format zurück auf die Diskette kopieren.

**Anmerkung:** Die Platzhalterzeichen \* und ? funktionieren mit den in diesem Abschnitt beschriebenen Befehlen nicht ordnungsgemäß (auch wenn sie in der BOS-Shell verwendet werden können). Wenn Sie keine Dateierweiterung angeben, wird der Dateiname so behandelt, als sei eine Erweiterung mit Leerzeichen angegeben worden.

### DOS-Dateien in Dateien des Basisbetriebssystems kopieren:

Mit dem Befehl **dosread** können Sie die angegebene DOS-Datei in die angegebene Datei des Basisbetriebssystems kopieren.

**Anmerkung:** Mit einer Ausnahme werden die Namenskonventionen von DOS verwendet. Da der Backslash (\) im Basisbetriebssystem eine Sonderbedeutung haben kann, müssen Sie in DOS-Pfadnamen den Schrägstrich (/) als Begrenzer für die Angabe von Unterverzeichnisnamen verwenden.

Beispiele:

- Wenn Sie eine Textdatei mit dem Namen chap1.doc von einer DOS-Diskette in das Basisbetriebsdateisystem kopieren möchten, geben Sie Folgendes ein:

```
dosread -a chap1.doc chap1
```

Dieser Befehl kopiert die DOS-Textdatei \CHAP1.DOC auf der Standardeinheit /dev/fd0 in die Basisbetriebssystemdatei chap1 im aktuellen Verzeichnis.

- Wenn Sie eine Binärdatei von einer DOS-Diskette in das Basisbetriebsdateisystem kopieren möchten, geben Sie Folgendes ein:

```
dosread -D/dev/fd0 /survey/test.dta /home/fran/testdata
```

Dieser Befehl kopiert die DOS-Datendatei \SURVEY\TEST.DTA auf /dev/fd0 in die Basisbetriebssystemdatei /home/fran/testdata.

Die vollständige Syntax des Befehls **dosread** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

### Dateien des Basisbetriebssystems in DOS-Dateien kopieren:

Mit dem Befehl **doswrite** können Sie die angegebene Datei des Basisbetriebssystems in die angegebene DOS-Datei kopieren.

**Anmerkung:** Mit einer Ausnahme werden die Namenskonventionen von DOS verwendet. Da der Backslash (\) im Basisbetriebssystem eine Sonderbedeutung haben kann, müssen Sie in DOS-Pfadnamen den Schrägstrich (/) als Begrenzer für die Angabe von Unterverzeichnisnamen verwenden.

Beispiele:

- Wenn Sie eine Textdatei mit dem Namen chap1 aus dem Basisbetriebsdateisystem auf eine DOS-Diskette kopieren möchten, geben Sie Folgendes ein:

```
doswrite -a chap1 chap1.doc
```

Dieser Befehl kopiert die Basisbetriebssystemdatei chap1 aus dem aktuellen Verzeichnis in die DOS-Textdatei \CHAP1.DOC auf /dev/fd0.

- Wenn Sie eine Binärdatei mit dem Namen /survey/test.dta aus dem Basisbetriebsdateisystem auf eine DOS-Diskette kopieren möchten, geben Sie Folgendes ein:

```
doswrite -D/dev/fd0 /home/fran/testdata /survey/test.dta
```

Dieser Befehl kopiert die Datendatei /home/fran/testdata des Basisbetriebssystems in die DOS-Datei \SURVEY\TEST.DTA auf /dev/fd0.

Die vollständige Syntax des Befehls **doswrite** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

### DOS-Dateien löschen:

Mit dem Befehl **dosdel** können Sie die angegebene DOS-Datei löschen.

**Anmerkung:** Mit einer Ausnahme werden die Namenskonventionen von DOS verwendet. Da der Backslash (\) im Basisbetriebssystem eine Sonderbedeutung haben kann, müssen Sie in DOS-Pfadnamen den Schrägstrich (/) als Begrenzer für die Angabe von Unterverzeichnisnamen verwenden.

Der Befehl **dosdel** konvertiert Kleinbuchstaben im Datei- oder Verzeichnisnamen in Großbuchstaben, bevor die Diskette geprüft wird. Da angenommen wird, dass es sich bei allen Pfadnamen um vollständige und nicht um relative Pfadnamen handelt, muss der erste Schrägstrich (/) nicht eingegeben werden.

Wenn Sie beispielsweise eine DOS-Datei mit dem Namen file.ext auf der Standardeinheit (/dev/fd0) löschen möchten, geben Sie Folgendes ein:

```
dosdel file.ext
```

Die vollständige Syntax des Befehls **dosdel** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

### Inhalt eines DOS-Verzeichnisses anzeigen:

Mit dem Befehl **dosdir** können Sie Informationen zu den angegebenen DOS-Dateien oder -Verzeichnissen anzeigen.

**Anmerkung:** Mit einer Ausnahme werden die Namenskonventionen von DOS verwendet. Da der Backslash (\) im Basisbetriebssystem eine Sonderbedeutung haben kann, müssen Sie in DOS-Pfadnamen den Schrägstrich (/) als Begrenzer für die Angabe von Unterverzeichnisnamen verwenden.

Der Befehl **dosdir** konvertiert Kleinbuchstaben im Datei- oder Verzeichnisnamen in Großbuchstaben, bevor die Diskette geprüft wird. Da angenommen wird, dass es sich bei allen Pfadnamen um vollständige und nicht um relative Pfadnamen handelt, muss der erste Schrägstrich (/) nicht eingegeben werden.

Wenn Sie beispielsweise ein Verzeichnis mit den DOS-Dateien auf der Einheit /dev/fd0 einlesen möchten, geben Sie den folgenden Befehl ein:

```
dosdir
```

Die Namen der Dateien und Informationen zum Plattenspeicherplatz werden in etwa wie folgt ausgegeben.

```
PG3-25.TXT
PG4-25.TXT
PG5-25.TXT
PG6-25.TXT
Free space: 312320 bytes
```

Die vollständige Syntax des Befehls **dosdir** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.



## Befehlsübersicht für Dateien

Im Folgenden finden Sie eine Übersicht über die Befehle für Dateien, Dateiverwaltungsprozeduren und DOS-Dateien. Außerdem enthält dieser Abschnitt eine Liste mit Befehlen für das Verknüpfen von Dateien und Verzeichnissen.

*Tabelle 57. Befehle für Dateien*

Eintrag	Beschreibung
*	Platzhalterzeichen, ersetzt beliebige Zeichenfolgen.
?	Platzhalterzeichen, ersetzt einzelne Zeichen.
[ ]	Metazeichen, ersetzt in Klammern eingeschlossene Zeichen.

*Tabelle 58. Befehle für Dateiverwaltungsprozeduren*

Eintrag	Beschreibung
cat	Verknüpft Dateien oder zeigt sie an.
cmp	Vergleicht zwei Dateien.
colrm	Extrahiert Spalten aus einer Datei.
cp	Kopiert Dateien.
cut	Gibt ausgewählte Bytes, Zeichen oder Felder aus jeder Zeile einer Datei aus.
diff	Vergleicht Textdateien.
file	Ermittelt den Dateityp.
find	Sucht Dateien unter Verwendung eines Abgleichsausdrucks.
grep	Durchsucht Dateien nach einem Muster.
head	Zeigt die ersten Zeilen oder Bytes einer oder mehrerer Datei(en) an.
more	Zeigt fortlaufenden Text seitenweise am Bildschirm an.
mv	Verschiebt Dateien.
nl	Nummeriert die Zeilen in einer Datei.
pg	Formatiert Dateien in der Anzeige.
rm	Löscht bzw. hebt die Verbindung zu Dateien oder Verzeichnissen auf.
paste	Verknüpft die Zeilen mehrerer Dateien oder aufeinanderfolgende Zeilen zu einer Datei.
sort	Sortiert Dateien, mischt bereits sortierte Dateien und überprüft, ob Dateien bereits sortiert sind.
tail	Schreibt eine Datei ab einer angegebenen Stelle in die Standardausgabe.
wc	Zählt die Anzahl der Zeilen, Wörter und Bytes in einer Datei.

*Tabelle 59. Befehle für das Verknüpfen von Dateien und Verzeichnissen*

Eintrag	Beschreibung
ln	Verbindet Dateien und Verzeichnisse.

*Tabelle 60. Befehle für DOS-Dateien*

Eintrag	Beschreibung
dosdel	Löscht DOS-Dateien.
dosdir	Listet das Verzeichnis für DOS-Dateien auf.
dosread	Kopiert DOS-Dateien in Dateien des Basisbetriebssystems.
doswrite	Kopiert Dateien des Basisbetriebssystems in DOS-Dateien.

## Betriebssystem-Shells

Die Schnittstelle zum Betriebssystem wird als *Shell* bezeichnet.

Die Shell ist die äußere Schicht des Betriebssystems. Shells besitzen eine Programmiersprache für die Steuerung von Prozessen und Dateien sowie für das Starten und Steuern anderer Programme. Die Shell verwaltet die Interaktion zwischen dem Benutzer und dem Betriebssystem, indem es den Benutzer zur Eingabe auffordert, diese Eingabe für das Betriebssystem interpretiert und anschließend die Ausgabe des Betriebssystems verarbeitet.

Shells bieten dem Benutzer eine Möglichkeit zur Kommunikation mit dem Betriebssystem. Diese Kommunikation erfolgt entweder interaktiv (auf Tastatureingaben erfolgt sofort eine Reaktion) oder mit einem Shell-Script. Ein *Shell-Script* ist eine Folge von Shell- und Betriebssystembefehlen, die in einer Datei gespeichert sind.

Wenn sich der Benutzer am System anmeldet, sucht das System den Namen eines Shellprogramms, um dieses auszuführen. Nach der Ausführung zeigt die Shell eine Eingabeaufforderung an. Diese Eingabeaufforderung ist normalerweise ein Dollarzeichen (\$). Wenn Sie an der Eingabeaufforderung einen Befehl eingeben und die Eingabetaste drücken, wertet die Shell den Befehl erst aus und versucht, ihn auszuführen. Abhängig von den Anweisungen des Befehls, wird die Ausgabe von der Shell am Bildschirm angezeigt oder umgeleitet. Dann gibt die Shell die Eingabeaufforderung zurück und erwartet von Ihnen die Eingabe eines weiteren Befehls.

In der *Befehlszeile* geben Sie Ihre Befehle ein. Diese Zeile enthält die Shellingabeaufforderung. Das Basisformat für jede Zeile ist:

```
$ Befehlsargument(e)
```

Die Shell interpretiert das erste Wort einer Befehlszeile (bis zum ersten Leerzeichen) als Befehl und alle nachfolgenden Wörter als Argumente.

**Anmerkung:** Wenn `libc.a` verschoben oder unbenannt wird, gibt die Shell die Fehlermeldung `Killed` aus, weil keine Datei `libc.a` verfügbar ist, die das System laden und für die Ausführung der Dienstprogramme verwenden könnte. Der Befehl `recsh` ruft die Shell für Wiederherstellung auf, in der Sie eine Datei `libc.a` umbenennen können, falls diese versehentlich verschoben wurde.

#### Zugehörige Tasks:

„Zuvor eingegebene Befehle auflisten (Befehl history)“ auf Seite 137

Mit dem Befehl `history` können Sie die Befehle auflisten, die Sie zuvor eingegeben haben.

## Shellkonzepte

Bevor Sie mit der Verwendung der verschiedenen Typen von Shells beginnen, die in AIX verfügbar sind, müssen Sie sich mit der grundlegenden Terminologie und den Basisfeatures vertraut machen.

#### Verfügbare Shells:

Die folgenden Shells werden mit AIX bereitgestellt.

- Korn-Shell (wird mit dem Befehl `ksh` gestartet)
- Bourne-Shell (wird mit dem Befehl `bsh` gestartet)
- Restricted Shell (eine eingeschränkte Version der Bourne-Shell, die mit dem Befehl `rsh` gestartet wird)
- POSIX-Shell (oder Korn-Shell, wird mit dem Befehl `psh` gestartet)
- Restricted Shell für die Korn-Shell (`ksh` und `ksh93`). Die Shells `ksh` und `ksh93` werden mit ihren Restricted-Shellentsprechungen `rksh` und `rksh93` bereitgestellt.
- Standardshell (wird mit dem Befehl `sh` gestartet)
- C-Shell (wird mit dem Befehl `csh` gestartet)
- Trusted Shell (eine eingeschränkte Version der Korn-Shell, die mit dem Befehl `tsh` gestartet wird)
- Ferne Shell (wird mit dem Befehl `rsh` gestartet)

Die *Anmeldeshell* ist die Shell, die geladen wird, wenn der Benutzer sich am System anmeldet. Die *Anmeldeshell* wird in der Datei `/etc/passwd` festgelegt. Die Korn-Shell ist die Standardanmeldeshell des Betriebssystems. Sie ist abwärtskompatibel mit der Bourne-Shell.

Die Korn-Shell (`/usr/bin/ksh`) wird als Standardshell eingerichtet. Die Standardshell ist die Shell, zu der eine Verbindung hergestellt und die mit dem Befehl `/usr/bin/sh` gestartet wird. Die Bourne-Shell (`/usr/`

bin/sh) kann als Standardshell eingesetzt werden. Die POSIX-Shell, die mit dem Befehl /usr/bin/psh aufgerufen wird, ist eine Verbindung zum Befehl /usr/bin/sh.

### Zugehörige Konzepte:

„Bourne-Shell“ auf Seite 271

Die Bourne-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlsorientierte Programmiersprache.

„Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 264

Die Korn-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlsorientierte Programmiersprache. Sie entspricht dem internationalen Standard für Betriebssysteme POSIX (Portable Operating System Interface for Computer Environment).

### Shellterminologie:

Die Begriffe und Definitionen in dieser Tabelle sind hilfreich für das Verständnis von Shells.

Eintrag	Beschreibung
<b>Leerzeichen</b>	Ein Leerzeichen ist eines der Zeichen der Leerzeichenklasse, die in der Kategorie <code>LC_CTYPE</code> definiert sind. In der POSIX-Shell ist ein Leerzeichen entweder ein Tabulator oder ein Leerschritt.
<b>Integrierter Befehl</b>	Ein Befehl, den die Shell ausführt, ohne ihn zu suchen und einen gesonderten Prozess zu erzeugen.
<b>Befehl</b>	Eine Folge von Zeichen in der Syntax der Shell-Sprache. Die Shell liest jeden Befehl und führt die gewünschte Aktion entweder direkt oder durch den Aufruf gesonderter Dienstprogramme aus.
<b>Kommentar</b>	Als Kommentar wird jedes Wort bezeichnet, das mit dem Nummernzeichen (#) beginnt. Dieses Wort und alle folgenden Zeichen bis zum nächsten Zeilenvorschubzeichen werden ignoriert.
<b>Kennung</b>	Eine Folge von Buchstaben, Ziffern oder Unterstreichungszeichen des portierbaren Zeichensatzes, die mit einem Unterstreichungszeichen oder einem Buchstaben beginnt. Das erste Zeichen einer Kennung darf keine Ziffer sein. Kennungen werden als Namen für Aliasnamen, Funktionen und als benannter Parameter verwendet.
<b>Liste</b>	<p>Eine Folge von Befehlsketten, die durch eines der folgenden Symbole voneinander getrennt werden: Semikolon (;), Et-Zeichen (&amp;), doppeltes Et-Zeichen (&amp;&amp;) oder doppeltes Pipe-Zeichen (   ). Die Liste kann optional mit einem der folgenden Zeichen beendet werden: Semikolon (;), Et-Zeichen (&amp;) oder Pipe-Zeichen mit Et-Zeichen ( &amp;).</p> <p>;</p> <p>Führt die vorhergehende Befehlskette in der angegebenen Reihenfolge aus. Die Shell führt alle Befehle der Reihe nach aus und wartet, bis der letzte Befehl abgeschlossen ist.</p> <p>&amp;</p> <p>Führt die vorhergehende Befehlskette asynchron aus. Die Shell führt alle Befehle der Reihe nach aus, verarbeitet die Befehlskette im Hintergrund und wartet nicht, bis der Befehl abgeschlossen ist.</p> <p> &amp;</p> <p>Führt die vorhergehende Befehlskette asynchron aus und richtet eine bidirektionale Befehlskette zur übergeordneten Shell ein. Die Shell führt alle Befehle der Reihe nach aus, verarbeitet die Befehlskette im Hintergrund und wartet nicht, bis der Befehl abgeschlossen ist. Die übergeordnete Shell kann mithilfe der Befehle <code>read -p</code> und <code>print -p</code> Lese- bzw. Schreiboperationen mit der Standardein-/ausgabe des erstellten Befehls durchführen. Zum jeweiligen Zeitpunkt kann nur ein derartiger Befehl aktiv sein.</p> <p>&amp;&amp;</p> <p>Verarbeitet die Liste, die diesem Symbol folgt, nur dann, wenn die vorhergehende Befehlskette einen Exit-Wert gleich null (0) zurückgibt.</p> <p>   </p> <p>Verarbeitet die Liste, die diesem Symbol folgt, nur dann, wenn die vorherige Befehlskette einen Exit-Wert ungleich null zurückgibt.</p> <p>Das Semikolon (;), das Et-Zeichen (&amp;) und das Pipe-Zeichen mit Et-Zeichen ( &amp;) haben eine niedrigere Priorität als das doppelte Et-Zeichen (&amp;&amp;) und das doppelte Pipe-Zeichen (   ). Die Symbole ;, &amp; und  &amp; sind untereinander gleichwertig. Dies gilt auch für die Symbole &amp;&amp; und    . Ein oder mehrere Zeilenvorschubzeichen können anstelle eines Semikolons verwendet werden, um zwei Befehle in einer Liste voneinander abzugrenzen.</p> <p><b>Anmerkung:</b> Das Symbol  &amp; ist nur in der Korn-Shell gültig.</p>

Eintrag	Beschreibung
<b>Metazeichen</b>	Jedes Metazeichen hat in der Shell eine Sonderbedeutung und bewirkt die Beendigung eines Wortes, sofern es nicht in Anführungszeichen gesetzt ist. Metazeichen sind das Pipe-Zeichen ( <code> </code> ), das Et-Zeichen ( <code>&amp;</code> ), das Semikolon ( <code>;</code> ), das Kleiner-als-Zeichen ( <code>&lt;</code> ), das Größer-als-Zeichen ( <code>&gt;</code> ), die linke runde Klammer ( <code>(</code> ), die rechte runde Klammer ( <code>)</code> ), das Dollarzeichen ( <code>\$</code> ), das umgekehrte Anführungszeichen ( <code>`</code> ), der Backslash ( <code>\</code> ), das rechte Anführungszeichen ( <code>'</code> ), das doppelte Anführungszeichen ( <code>"</code> ), das Zeilenvorschubzeichen, das Leerzeichen und das Tabulatorzeichen. Alle Zeichen, die in einfache Anführungszeichen eingeschlossen sind, werden als Zitat eingestuft und von der Shell wörtlich interpretiert. Die Sonderbedeutung von Metazeichen bleibt erhalten, wenn keine Anführungszeichen verwendet werden. (Metazeichen werden in der C-Shell auch als <i>Parser-Metazeichen</i> bezeichnet.)
<b>Parameterzuordnungsliste</b>	Enthält ein oder mehrere Wörter im Format <i>ID=Wert</i> , wobei vor und hinter dem Gleichheitszeichen (=) dieselbe Anzahl von Leerzeichen verwendet werden muss. Dies bedeutet, dass entweder führende und folgende Leerzeichen oder überhaupt keine Leerzeichen verwendet werden müssen. <b>Anmerkung:</b> In der C-Shell hat die Parameterzuordnungsliste das Format <i>setID=Wert</i> . Die Leerzeichen vor und hinter dem Gleichheitszeichen (=) sind erforderlich.
<b>Befehlskette</b>	Eine Folge von Befehlen, die durch das Pipe-Symbol ( <code> </code> ) voneinander getrennt sind. Jeder Befehl in der Befehlskette, möglicherweise mit Ausnahme des letzten Befehls, wird separat verarbeitet. Die Standardausgabe jedes Befehls in einer Befehlskette wird jedoch als Standardeingabe für den nächsten Befehl in der Folge verwendet. Wenn eine Liste in runde Klammern eingeschlossen ist, wird sie als einfacher Befehl in einer separaten Subshell ausgeführt.  Falls der Befehlskette das reservierte Wort <code>!</code> nicht vorangestellt ist, entspricht der Exit-Status dem Exit-Status des vorherigen in der Befehlskette angegebenen Befehls. Andernfalls ist der Exit-Status das logische NICHT des Exit-Status des vorherigen Befehls. Mit anderen Worten, wenn der vorherige Befehl null zurückgibt, ist der Exit-Status 1. Wenn der vorherige Befehl einen Wert größer als null zurückgibt, ist der Exit-Status null.  Das Format einer Befehlskette ist wie folgt: <code>[!] Befeh1 [   Befeh2 ...]</code>  <b>Anmerkung:</b> In früheren Versionen der Bourne-Shell wurde das Winkelzeichen ( <code>^</code> ) zur Kennzeichnung einer Pipe verwendet.
<b>Shellvariable</b>	Ein Name oder Parameter, dem ein Wert zugeordnet wird. Für die Zuordnung einer Variablen geben Sie den Variablennamen, ein Gleichheitszeichen (=) und dann den Wert ein. Der Variablenname kann den zugeordneten Wert ersetzen, indem dem Variablennamen ein Dollarzeichen (\$) vorangestellt wird. Variablen sind besonders hilfreich bei der Erstellung einer Abkürzung für einen langen Pfadnamen, wie z. B. <code>\$HOME</code> für das Ausgangsverzeichnis. Eine vordefinierte Variable ist eine Variable, deren Wert von der Shell zugeordnet wird. Eine benutzerdefinierte Variable ist eine Variable, deren Wert vom Benutzer zugeordnet wird.
<b>Einfacher Befehl</b>	Eine Folge optionaler Parameterzuordnungslisten und Umleitungen in einer beliebigen Reihenfolge. Ihnen können Befehle, Wörter und Umleitungen folgen. Sie werden mit einem Semikolon ( <code>;</code> ), Pipe-Zeichen ( <code> </code> ), Et-Zeichen ( <code>&amp;</code> ), doppelten Pipe-Zeichen ( <code>  </code> ), doppelten Et-Zeichen ( <code>&amp;&amp;</code> ), Pipe-Zeichen mit Et-Zeichen <code> &amp;</code> oder einem Zeilenvorschubzeichen beendet. Der Befehlsname wird (gemäß Definition der Subroutine <code>exec</code> ) als Parameter <code>0</code> übergeben. Der Wert eines einfachen Befehls ist ein Exit-Status gleich null, wenn er normal beendet wird, bzw. ein Exit-Status ungleich null, wenn er mit Fehlern beendet wird. Die Subroutinen <code>sigaction</code> , <code>sigvec</code> und <code>signal</code> enthalten eine Liste mit den Exit-Status für Signale.
<b>Subshell</b>	Eine Shell, die als untergeordnete Shell der Anmeldeshell oder der aktuellen Shell ausgeführt wird.
<b>Platzhalterzeichen</b>	Die Shell weist diesen Platzhalterzeichen zugeordnete Werte zu. Die Basisplatzhalterzeichen sind <code>?</code> , <code>*</code> , <code>[set]</code> und <code>!set</code> . Platzhalterzeichen sind besonders bei der Dateinamensubstitution hilfreich.
<b>Wort</b>	Eine Folge von Zeichen, die keine Leerzeichen enthält. Wörter werden durch ein oder mehrere Metazeichen voneinander getrennt.

### Shell für eine Scriptdatei angeben:

Wenn Sie ein ausführbares Shell-Script in der Korn- (bzw. POSIX-Shell) oder Bourne-Shell starten, werden die Befehle im Script unter der Steuerung der aktuellen Shell ausgeführt, d. h. der Shell, in der Sie das

Script gestartet haben, sofern Sie keine andere Shell angeben. Wenn Sie ein Shell-Script in der C-Shell starten, werden die Befehle im Script unter der Steuerung der Bourne-Shell (`/usr/bin/bsh`) ausgeführt, sofern Sie keine andere Shell angeben.

Durch Angabe der Shell im Shell-Script können Sie ein Shell-Script in einer bestimmten Shell ausführen.

Wenn Sie ein ausführbares Shell-Script in einer bestimmten Shell ausführen möchten, geben Sie in der ersten Zeile des Shell-Script `#!/Pfad` ein, und drücken Sie anschließend die Eingabetaste. Die Zeichen `#!` kennzeichnen den Dateityp. Die Variable `Pfad` gibt den Pfadnamen der Shell an, in der das Shell-Script ausgeführt werden soll.

Geben Sie beispielsweise Folgendes ein, um das Script `bsh` in der Bourne-Shell auszuführen:

```
#!/usr/bin/bsh
```

Wenn Sie vor dem Namen der Shell-Scriptdatei einen Shellbefehl angeben, setzt die in der Befehlszeile angegebene Shell die in der Scriptdatei angegebene Shell außer Kraft. Wenn Sie also `ksh meinedatei` eingeben und die Eingabetaste drücken, wird die Datei `meinedatei` unter der Steuerung der Korn-Shell ausgeführt, selbst wenn die Datei `meinedatei` in der ersten Zeile `#!/usr/bin/csh` enthält.

### Shellfunktionen:

Die Verwendung der Shell als Schnittstelle zum System hat verschiedene Vorteile.

Im Folgenden sind die Hauptvorteile der Systembedienung über eine Shell aufgeführt:

- **Verwendung von Platzhalterzeichen in Dateinamen (Mustererkennung)**

Führt Befehle für eine Dateigruppe aus, die mit einem Erkennungsmuster und nicht mit einem tatsächlichen Dateinamen angegeben werden.

Nähere Informationen hierzu finden Sie in den folgenden Abschnitten:

- „Dateinamenssubstitution in der Korn- bzw. POSIX-Shell“ auf Seite 242
- „Dateinamenssubstitution in der Bourne-Shell“ auf Seite 274
- „Dateinamenssubstitution in der C-Shell“ auf Seite 292

- **Hintergrundverarbeitung**

Definiert die Ausführung komplexer Tasks im Hintergrund und macht auf diese Weise das Terminal für parallele Dialogverarbeitung frei.

Nähere Informationen hierzu finden Sie in der Beschreibung des Befehls `bg` in den folgenden Abschnitten:

- „Jobsteuerung in der Korn- bzw. POSIX-Shell“ auf Seite 256
- „Integrierte C-Shellbefehle“ auf Seite 299

**Anmerkung:** Die Bourne-Shell unterstützt keine Jobsteuerung.

- **Befehls-Aliasing**

Vergibt einen Aliasnamen für einen Befehl oder Ausdruck. Wenn die Shell in der Befehlszeile oder in einem Shell-Script einen Aliasnamen erkennt, ersetzt sie diesen durch den Text, der dem Aliasnamen zugeordnet ist.

Nähere Informationen hierzu finden Sie in den folgenden Abschnitten:

- „Befehls-Aliasing in der Korn- bzw. POSIX-Shell“ auf Seite 270
- „Aliassubstitution in der C-Shell“ auf Seite 289

**Anmerkung:** Die Bourne-Shell unterstützt kein Befehls-Aliasing.

- **Befehlsprotokoll**

Zeichnet die eingegebenen Befehle in einer Protokolldatei auf. Diese Datei kann verwendet werden, um schnell auf einen aufgelisteten Befehl zuzugreifen, ihn zu verändern und erneut auszugeben.

Nähere Informationen hierzu finden Sie in der Beschreibung des Befehls **history** in den folgenden Abschnitten:

- „Befehlsprotokoll in der Korn- bzw. POSIX-Shell“ auf Seite 269
- „Integrierte C-Shellbefehle“ auf Seite 299
- „Protokollsubstitution in der C-Shell“ auf Seite 307

**Anmerkung:** Die Bourne-Shell unterstützt keine Befehlsprotokolle.

- **Dateinamensubstitution**

Erzeugt automatisch mithilfe von Mustererkennungszeichen eine Liste mit Dateinamen in einer Befehlszeile.

Nähere Informationen hierzu finden Sie in den folgenden Abschnitten:

- „Dateinamensubstitution in der Korn- bzw. POSIX-Shell“ auf Seite 242
- „Dateinamensubstitution in der Bourne-Shell“ auf Seite 274
- „Dateinamensubstitution in der C-Shell“ auf Seite 292

- **Eingabe- und Ausgabeumleitung**

Eingabeumleitung von der Tastatur und Ausgabeumleitung in eine Datei oder an eine andere Einheit als das Terminal. Eingaben in ein Programm können von einer Datei bereitgestellt und an den Drucker oder in eine andere Datei umgeleitet werden.

Nähere Informationen hierzu finden Sie in den folgenden Abschnitten:

- „Eingabe- und Ausgabeumleitung in der Korn- bzw. POSIX-Shell“ auf Seite 243
- „Eingabe- und Ausgabeumleitung in der Bourne-Shell“ auf Seite 275
- „Eingabe- und Ausgabeumleitung in der C-Shell“ auf Seite 310

- **Befehlsketten**

Verbinden eine beliebige Anzahl von Befehlen zu einem komplexen Programm. Die Standardausgabe eines Programms wird zur Standardeingabe des folgenden Programms.

Nähere Informationen hierzu finden Sie in der Definition von *Befehlsketten* im Abschnitt „Shellterminologie“ auf Seite 219.

- **Shellvariablensubstitution**

Speichert Daten in benutzerdefinierten Variablen und vordefinierten Shellvariablen.

Nähere Informationen hierzu finden Sie in den folgenden Abschnitten:

- „Parametersubstitution in der Korn- bzw. POSIX-Shell“ auf Seite 240
- „Variablensubstitution in der Bourne-Shell“ auf Seite 284
- „Variablensubstitution in der C-Shell“ auf Seite 290

**Zugehörige Konzepte:**

„Befehle“ auf Seite 133

Manche Befehle können durch Eingabe eines einzigen Wortes ausgeführt werden. Es ist auch möglich, Befehle zu kombinieren, um die Ausgabe eines Befehls als Eingabe für einen anderen Befehl zu verwenden.

**Zeichenklassen:**

Sie können Zeichenklassen für den Abgleich von Dateinamen verwenden.

Sie können für den Abgleich von Dateinamen wie folgt Zeichenklassen verwenden:

`[[:Zeichenklasse:]]`

Mit diesem Format wird das System angewiesen, jedes einzelne Zeichen, das zur angegebenen Klasse gehört, abzugleichen. Die definierten Klassen entsprechen den Subroutinen **ctype**.

Zeichenklasse	Definition
<b>alnum</b>	Alphanumerische Zeichen
<b>alpha</b>	Groß- und Kleinbuchstaben
<b>blank</b>	Leerzeichen oder Horizontaltabulator
<b>cntrl</b>	Steuerzeichen
<b>digit</b>	Ziffern
<b>graph</b>	Grafikzeichen
<b>lower</b>	Kleinbuchstaben
<b>print</b>	Druckbare Zeichen
<b>punct</b>	Interpunktionszeichen
<b>space</b>	Leerzeichen, Horizontaltabulatorzeichen, Wagenrücklaufzeichen, Zeilenvorschubzeichen, Vertikaltabulatorzeichen oder Formularvorschubzeichen
<b>upper</b>	Großbuchstaben
<b>xdigit</b>	Hexadezimalzeichen

### Restricted Shell:

Mit der Restricted Shell (eingeschränkte Shell) werden Anmeldenamen und Ausführungsumgebungen eingerichtet, deren Leistungsspektrum geringer ist als das der regulären Bourne-Shell.

Die Restricted Shell wird mit dem Befehl **rsh** oder **bsh -r** geöffnet. Die Aktionen dieser Befehle entsprechen im Wesentlichen denen des Befehls **bsh**. Eine Ausnahme bilden die folgenden Aktionen, die in der Restricted Shell nicht zulässig sind:

- Verzeichnis wechseln (mit dem Befehl **cd**)
- Werte der Variablen *PATH* und *SHELL* festlegen
- Pfad- oder Befehlsnamen, die einen Schrägstrich (/) enthalten, angeben
- Ausgabe umleiten

Stellt die Restricted Shell fest, dass es sich bei einem auszuführenden Befehl um eine Shellprozedur handelt, ruft sie zur Ausführung des Befehls die Bourne-Shell auf. Auf diese Weise ist es möglich, dem Benutzer über ein eingeschränktes Befehlsmenü Shellprozeduren zur Verfügung zu stellen, die auf das gesamte Leistungsspektrum der Bourne-Shell zugreifen. In dieser Situation wird davon ausgegangen, dass der Benutzer nicht über Schreib- und Ausführungsberechtigungen in demselben Verzeichnis verfügt.

Wird der Parameter *Datei* [*Parameter*] beim Aufruf der Bourne-Shell angegeben, führt sie die im Parameter *Datei* angegebene Scriptdatei mit allen angegebenen Parametern aus. Die angegebene Scriptdatei muss über Lesezugriff verfügen. Die **setuid**- und **setgid**-Einstellungen für Scriptdateien werden ignoriert. Anschließend liest die Shell die Befehle. Geben Sie keine Scriptdatei an, wenn Sie das Flag **-c** oder **-s** verwenden.

Wird die Shell mit dem Befehl **rsh** gestartet, führt die Shell nach der Interpretation der Dateien *.profile* und */etc/environment* Einschränkungen ein. Deshalb kann der Autor der Datei *.profile* Benutzeraktionen genau steuern, indem er Konfigurationsaktionen ausführt und den Benutzer in ein bestimmtes Verzeichnis (möglicherweise nicht das Anmeldeverzeichnis) stellt. Ein Administrator kann durch entsprechende Änderung der Variablen *PATH* ein Verzeichnis mit Befehlen im Verzeichnis */usr/sbin* einrichten, das vom Befehl **rsh** verwendet werden kann. Wird die Shell mit dem Befehl **bsh -r** aufgerufen, führt sie die Einschränkungen bei der Interpretation der Dateien *.profile* ein.

Wird die Restricted Shell unter dem Namen **rsh** aufgerufen, liest sie die Datei *.profile* des Benutzers (*\$HOME/.profile*). Sie verhält sich dabei wie eine reguläre Bourne-Shell, mit der Ausnahme, dass die Shell durch eine Unterbrechung sofort verlassen und keine Rückkehr auf Befehlsebene erfolgt.

Die Korn-Shell kann mit dem Befehl **ksh -r** als Restricted Shell gestartet werden.

Die I-Nodes für **ksh** und **rksh** sind identisch, ebenso die I-Nodes für **ksh93** und **rksh93**.

### Shell-Script erstellen und ausführen:

Ein *Shell-Script* ist eine Datei, die einen oder mehrere Befehl(e) enthält. Shell-Scripts sind eine einfache Möglichkeit, komplexe Befehle, lange oder komplizierte Befehlsfolgen und Routineaufgaben auszuführen. Wenn der Benutzer den Namen einer Shell-Scriptdatei eingibt, führt das System die in der Datei enthaltene Befehlsfolge aus.

Ein Shell-Script kann in einem Texteditor erstellt werden. Das Script kann sowohl Befehle des Betriebssystems als auch integrierte Shellbefehle enthalten.

Im Folgenden finden Sie allgemeine Richtlinien für das Schreiben von Shell-Scripts:

1. Erstellen Sie in einem Texteditor eine Datei, und speichern Sie sie. Sie können beliebige Kombinationen von Shell- und Betriebssystembefehlen in die Datei für das Shell-Script aufnehmen. Es gilt die Konvention, dass Shell-Scripts, die nicht für mehrere Benutzer bestimmt sind, im Verzeichnis `$HOME/bin` gespeichert werden.

**Anmerkung:** Die Subroutinen **setuid** und **setgid** in einem Shell-Script werden vom Betriebssystem nicht unterstützt.

2. Verwenden Sie den Befehl **chmod**, wenn Sie festlegen möchten, dass nur der Eigner die Datei ausführen kann. Geben Sie beispielsweise folgenden Befehl ein, wenn die Datei den Namen `script1` hat:

```
chmod u=rwx script1
```

3. Geben Sie in der Befehlszeile den Scriptnamen ein, um das Shell-Script auszuführen. Zum Ausführen des Shell-Script `script1` geben Sie beispielsweise Folgendes ein:

```
script1
```

**Anmerkung:** Sie können ein Shell-Script ausführen, ohne es zu einer ausführbaren Datei zu machen, wenn Sie in der Befehlszeile dem Shellbefehl (**ksh**, **bsh** oder **cs**) den Namen der Shell-Scriptdatei voranstellen. Wenn Sie beispielsweise eine nicht ausführbare Datei mit dem Namen `script1` in der Korn-Shell ausführen möchten, geben Sie den folgenden Befehl ein:

```
ksh script1
```

### Zugehörige Konzepte:

„Befehle“ auf Seite 133

Manche Befehle können durch Eingabe eines einzigen Wortes ausgeführt werden. Es ist auch möglich, Befehle zu kombinieren, um die Ausgabe eines Befehls als Eingabe für einen anderen Befehl zu verwenden.

## Korn-Shell

Die Korn-Shell (Befehl `ksh`) ist mit der Bourne-Shell (Befehl `bsh`) abwärtskompatibel und enthält die meisten Features der Bourne-Shell sowie mehrere der nützlichsten Features der C-Shell.

### Von der Korn- bzw. POSIX-Shell gesetzte Variablen:

Die folgenden Variablen werden von der Shell gesetzt.



Eintrag	Beschreibung
<i>Unterstrichungszeichen</i> ( <u>  </u> )	Zeigt zunächst den absoluten Pfadnamen der Shell oder des Scripts an, die bzw. das sofort nach der Übergabe an die Umgebung ausgeführt wird. Nachfolgend wird das letzte Argument des vorherigen Befehls zugeordnet. Dieser Parameter wird nicht für asynchrone Befehle gesetzt. Außerdem enthält dieser Parameter den Namen der MAIL-Datei, wenn geprüft wird, ob Mails vorhanden sind.
<i>ERRNO</i>	Gibt den Wert an, der von der letzten gescheiterten Subroutine gesetzt wurde. Dieser Wert ist systemabhängig und dient zur Fehlerbehebung.
<i>LINENO</i>	Gibt die Zeilennummer der aktuellen Zeile im Script bzw. in der Funktion an, das bzw. die gerade ausgeführt wird.
<i>OLDPWD</i>	Zeigt das vorherige Arbeitsverzeichnis an, das durch den Befehl <b>cd</b> festgelegt wurde.
<i>OPTARG</i>	Gibt den Wert des letzten Optionsarguments an, das mit dem integrierten Befehl <b>getopts</b> verarbeitet wurde.
<i>OPTIND</i>	Gibt den Index des letzten Optionsarguments an, das mit dem integrierten Befehl <b>getopts</b> verarbeitet wurde.
<i>PPID</i>	Gibt die Prozessnummer des übergeordneten Prozesses der Shell an.
<i>PWD</i>	Zeigt das aktuelle Arbeitsverzeichnis an, das mit dem Befehl <b>cd</b> festgelegt wurde.
<i>RANDOM</i>	Generiert eine zufällige ganze Zahl zwischen 0 und 32767. Die Folge von Zufallszahlen kann durch die Zuordnung eines numerischen Werts zu der Variablen <i>RANDOM</i> initialisiert werden.
<i>REPLY</i>	Wird durch die Anweisung <b>select</b> und dem integrierten Befehl <b>read</b> gesetzt, wenn keine Argumente angegeben wurden.
<i>SECONDS</i>	Gibt die Anzahl an Sekunden an, die seit dem Aufruf der Shell vergangen sind. Wird dieser Variablen ein Wert zugeordnet, ist der hier zurückgegebene Wert der zugeordnete Wert plus die Anzahl an Sekunden seit der Zuordnung.

### Von der Korn- bzw. POSIX-Shell verwendete Variablen:

Die folgenden Variablen werden von der Shell verwendet.

Eintrag	Beschreibung
<i>CDPATH</i>	Zeigt den Suchpfad für den Befehl <b>cd</b> (change directory, Verzeichnis wechseln) an.
<i>COLUMNS</i>	Definiert die Breite des Editierfensters für die Shellvdiertiermodi und für die Ausgabe von Listen des Befehls <b>select</b> .
<i>EDITOR</i>	Endet der Wert dieses Parameters mit emacs, gmacs oder vi und wurde die Variable <i>VISUAL</i> nicht mit dem integrierten Sonderbefehl <b>set</b> gesetzt, wird die entsprechende Option aktiviert.
<i>ENV</i>	Ist diese Variable gesetzt, wird die Parametersubstitution für den Wert ausgeführt, um den Pfadnamen des Scripts zu generieren, das beim Aufruf der Shell ausgeführt wird. Diese Datei wird normalerweise für die Definition von Aliasnamen und Funktionen verwendet. Diese Variable wird in nicht interaktiven Shells ignoriert.
<i>FCEDIT</i>	Gibt den Standardeditornamen für den integrierten Befehl <b>fc</b> an.
<i>FPATH</i>	Gibt den Suchpfad für Funktionsdefinitionen an. Dieser Pfad wird durchsucht, wenn mit dem Flag <b>-u</b> auf eine Funktion verwiesen und kein Befehl gefunden wird. Wird eine ausführbare Datei gefunden, wird diese gelesen und in der aktuellen Umgebung ausgeführt.
<i>HISTFILE</i>	Ist diese Variable beim Aufruf der Shell gesetzt, ist ihr Wert der Pfadname der Datei, die zum Speichern des Befehlsprotokolls verwendet wird.  Der Initialisierungsprozess für die Datei <i>history</i> kann von den Startdateien des Systems abhängig sein, weil einige Startdateien Befehle enthalten können, die die vom Benutzer für die Variablen <i>HISTFILE</i> und <i>HISTSIZE</i> definierten Einstellungen überschreiben. Befehle für Funktionsdefinitionen werden beispielsweise in der Datei <i>history</i> aufgezeichnet. Wenn der Systemadministrator Funktionsdefinitionen in eine Systemstartdatei einfügt, die aufgerufen wird, bevor die Datei <i>ENV</i> aufgerufen wird oder bevor die Variable <i>HISTFILE</i> oder <i>HISTSIZE</i> gesetzt ist, wird die Datei <i>history</i> initialisiert, bevor der Benutzer Einfluss auf die Merkmale nehmen kann.

Eintrag	Beschreibung
<b>HISTSIZE</b>	Ist diese Variable beim Aufruf der Shell gesetzt, ist die Anzahl der zuvor eingegebenen Befehle, auf die diese Shell zugreifen kann, größer-gleich dieser Zahl. Standardmäßig werden die letzten 128 Befehle für andere Benutzer als Root und die letzten 512 Befehle für Root aufgezeichnet.
<b>HOME</b>	Zeigt den Namen des Anmeldeverzeichnisses an, das nach Abschluss der Anmeldung zum aktuellen Verzeichnis wird. Diese Variable wird vom Programm <b>login</b> initialisiert. Der Befehl <b>cd</b> verwendet den Wert des Parameters <i>\$HOME</i> als Standardwert. Durch die Verwendung dieser Variablen anstelle eines expliziten Pfadnamens in einer Shellprozedur kann die Prozedur ohne Änderungen aus verschiedenen Verzeichnissen gestartet werden.
<b>IFS</b>	Gibt interne Feldtrennzeichen (normalerweise Leerzeichen, Tabulator- und Zeilenvorschubzeichen) an, die verwendet werden, um Befehlswörter zu trennen, die aus der Befehls- oder Parametersubstitution resultieren, und um Wörter mit dem integrierten Befehl <b>read</b> zu trennen. Das erste Zeichen des Parameters <i>IFS</i> wird verwendet, um Argumente für die Substitution <i>\$*</i> zu trennen.
<b>LANG</b>	Setzt den Standardwert für die Variablen <i>LC_*</i> .
<b>LC_ALL</b>	Setzt den Wert der Variablen <i>LANG</i> und <i>LC_*</i> außer Kraft.
<b>LC_COLLATE</b>	Bestimmt die Funktionsweise von Bereichsausdrücken bei der Mustererkennung.
<b>LC_CTYPE</b>	Definiert die Zeichenklassifikation, den Wechsel zwischen Groß- und Kleinschreibung und andere Zeichenattribute.
<b>LC_MESSAGES</b>	Bestimmt, in welcher Sprache Nachrichten geschrieben werden.
<b>LINES</b>	Bestimmt die Spaltenlänge für die Ausgabe von Auswahllisten. Auswahllisten werden so ausgegeben, dass vertikal etwa zwei Drittel der in der Variablen <i>LINES</i> angegebenen Zeilen gefüllt sind.
<b>MAIL</b>	Gibt den Dateipfadnamen an, der vom Mail-System verwendet wird, um eingegangene Mails zu finden. Ist diese Variable auf den Namen einer Nachrichtendatei gesetzt, die Variable <i>MAILPATH</i> jedoch nicht gesetzt, informiert die Shell den Benutzer über den Eingang neuer Mails in der angegebenen Datei.
<b>MAILCHECK</b>	Gibt an, wie oft (in Sekunden) die Shell Veränderungen in der Änderungszeit bei den in der Variablen <i>MAILPATH</i> oder <i>MAIL</i> angegebenen Dateien überprüft. Der Standardwert sind 600 Sekunden. Nach Ablauf der Zeit führt die Shell vor der Ausgabe der nächsten Eingabeaufforderung eine Überprüfung durch.
<b>MAILPATH</b>	Gibt eine Liste mit Dateinamen an, die durch Doppelpunkte getrennt sind. Ist diese Variable gesetzt, informiert die Shell den Benutzer über alle Änderungen, die innerhalb des in der Variablen <i>MAILCHECK</i> angegebenen Berichtszeitraums (in Sekunden) an den angegebenen Dateien vorgenommen wurden. Jedem Dateinamen kann ein Fragezeichen <i>?</i> und eine Nachricht, die ausgegeben wird, folgen. Die Nachricht wird einer Variablensubstitution unterzogen, wobei die Variable <i>\$_</i> , die als Name der geänderten Datei definiert ist, verwendet wird. Die Standardnachricht ist lautet: <i>you have mail in \$_</i> .
<b>NLSPATH</b>	Legt die Position von Nachrichtenkatalogen für die Verarbeitung von <i>LC_MESSAGES</i> fest.
<b>PATH</b>	Zeigt den Suchpfad für Befehle an. Es handelt sich dabei um eine geordnete Liste mit Verzeichnispfadnamen, die durch Doppelpunkte getrennt sind. Die Shell durchsucht diese Verzeichnisse in der angegebenen Reihenfolge, wenn sie nach Befehlen sucht. Das aktuelle Verzeichnis wird in der Liste durch eine leere Zeichenfolge dargestellt.
<b>PS1</b>	Gibt die als Primäreingabeaufforderung zu verwendende Zeichenfolge an. Der Wert dieses Parameters wird für die Parametersubstitution erweitert, um die primäre Eingabeaufforderung zu definieren, die standardmäßig ein Dollarzeichen ( <i>\$</i> ) ist. Das Zeichen <i>!</i> in der primären Eingabeaufforderung wird durch die Befehlsnummer ersetzt.
<b>PS2</b>	Gibt den Wert der sekundären Eingabeaufforderung an, die standardmäßig durch <i>&gt;</i> dargestellt wird.
<b>PS3</b>	Gibt den Wert der Eingabeaufforderung für die Auswahl an, die innerhalb einer Schleife <b>select</b> verwendet und standardmäßig durch ein Nummernzeichen mit Fragezeichen <i>#?</i> dargestellt wird.
<b>PS4</b>	Der Wert dieser Variablen wird für die Parametersubstitution erweitert und jeder Zeile eines Ausführungstrace vorangestellt. Wird diese Variable nicht angegeben, ist die Eingabeaufforderung des Ausführungstrace ein Pluszeichen ( <i>+</i> ).
<b>SHELL</b>	Gibt den Pfadnamen der Shell an, die in der Umgebung enthalten ist.

Eintrag	Beschreibung
<b>SHELL PROMPT</b>	Im interaktiven Modus wird als Shellvingabeaufforderung der Wert des Parameters <i>PS1</i> angezeigt, bevor ein Befehl gelesen wird. Wenn eine neue Zeile eingegeben wird und die Shell weitere Eingaben benötigt, um einen Befehl auszuführen, gibt die Shell die sekundäre Eingabeaufforderung (den Wert des Parameters <i>PS2</i> ) aus.
<b>TMOU</b>	Gibt an, wie lange eine Shell inaktiv sein darf, bevor sie beendet wird. Wenn die Variable <i>TMOU</i> einen Wert größer als null hat, wird die Shell beendet, wenn innerhalb der vorgeschriebenen Anzahl von Sekunden nach Ausgabe der <i>PS1</i> -Eingabeaufforderung kein Befehl eingegeben wird. (Es ist zu beachten, dass die Shell mit einem maximalen Grenzwert kompiliert werden kann, der für den hier beschriebenen Wert nicht überschritten werden darf.) <b>Anmerkung:</b> Nach Ablauf des Zeitlimits erfolgt eine Pause von 60 Sekunden, bevor die Shell verlassen wird.
<b>VISUAL</b>	Wenn der Wert dieser Variablen mit <i>emacs</i> , <i>gmacs</i> oder <i>vi</i> endet, wird die entsprechende Option aktiviert.

Die Shell ordnet den Parametern *PATH*, *PS1*, *PS2*, *MAILCHECK*, *TMOU* und *IFS* Standardwerte zu. Die Parameter *HOME*, *SHELL*, *ENV* und *MAIL* werden jedoch *nicht* von der Shell gesetzt (obwohl der Parameter *HOME* vom Befehl **login** gesetzt wird).

### Befehlssubstitution in der Korn- bzw. POSIX-Shell:

Die Befehlssubstitution wird in der Korn- bzw. POSIX-Shell unterstützt. Bei der Befehlssubstitution führt die Shell den angegebenen Befehl in einer Subshell aus und ersetzt den Befehl durch seine Ausgabe.

Geben Sie Folgendes ein, um eine Befehlssubstitution in der Korn- bzw. POSIX-Shell durchzuführen:  
\$(Befehl)

Für die Befehlssubstitution können auch umgekehrte Anführungszeichen verwendet werden. Geben Sie dazu Folgendes ein:

```
`Befehl`
```

**Anmerkung:** Obwohl die Syntax mit umgekehrten Anführungszeichen in **ksh** akzeptiert wird, wird diese Schreibweise von den Standards X/Open Portability Guide Issue 4 und POSIX als veraltet betrachtet. Diese Standards empfehlen für portierbare Anwendungen die Verwendung der Syntax \$(Befehl).

Die Shell erweitert die Befehlssubstitution, indem sie den Befehl in einer SubShellumgebung ausführt und die Befehlssubstitution (den Text von Befehl einschließlich der umschließenden runden Klammern \$( )) oder der umgekehrten einfachen Anführungszeichen) durch die Standardausgabe des Befehls ersetzt und dabei Zeilenvorschubzeichen am Ende der Substitution entfernt.

Im folgenden Beispiel zeigen die Zeichen \$( ), die den Befehl umschließen, an, dass die Ausgabe des Befehls **whoami** ersetzt wird:

```
echo Mein Name ist: $(whoami)
```

Dieselbe Befehlssubstitution kann mit folgendem Befehl ausgeführt werden:

```
echo Mein Name ist: `whoami`
```

In beiden Beispielen würde die Ausgabe für Benutzer dagmar wie folgt lauten:

```
Mein Name ist: dagmar
```

Auch arithmetische Ausdrücke können durch das Einschließen in runde Klammern (( )) ersetzt werden. Beispielsweise führt der Befehl

```
echo Jede Stunde hat $((60 * 60)) Sekunden
```

zu folgendem Ergebnis:

Jede Stunde hat 3600 Sekunden

Die Korn- bzw. POSIX-Shell löscht bei der Befehlssubstitution alle abschließenden Zeilenvorschubzeichen. Enthält das aktuelle Verzeichnis beispielsweise die Dateien `Datei1`, `Datei2` und `Datei3`, löscht der Befehl `echo $(ls)`

die Zeilenvorschubzeichen und erzeugt folgende Ausgabe:

```
Datei1 Datei2 Datei3
```

Um Zeilenvorschubzeichen zu erhalten, muss der ersetzte Befehl in doppelte Anführungszeichen (" ") gesetzt werden:

```
echo "$(ls)"
```

### Arithmetische Berechnungen in der Korn- bzw. POSIX-Shell:

Mit dem integrierten Befehl `let` der Korn- bzw. POSIX-Shell können Sie Ganzzahlberechnungen durchführen.

Die Konstanten haben das Format **[Basis]Zahl**. Der Parameter **Basis** ist eine Dezimalzahl zwischen 2 und 36 einschließlich, die die arithmetische Basis angibt. Der Parameter **Zahl** ist eine Zahl zu dieser Basis. Wird der Parameter **Basis** nicht angegeben, verwendet die Shell die Basis 10.

Arithmetische Ausdrücke verwenden dieselbe Syntax, Vorrangstellung und Assoziativität wie die Programmiersprache C. Alle Integraloperatoren mit Ausnahme des doppelten Pluszeichens (`++`), des doppelten Minuszeichens (`--`), des Fragezeichens mit direkt anschließendem Doppelpunkt (`?:`) und des Kommas (`,`) werden unterstützt. In der folgenden Tabelle sind die gültigen Operatoren für die Korn- bzw. POSIX-Shell in absteigender Reihenfolge ihrer Priorität aufgelistet:

Operator	Definition
-	Unäres Minus
!	Logische Negation
~	Bitweise Negation
*	Multiplikation
/	Division
%	Rest
+	Addition
-	Subtraktion
<<, >>	Verschieben nach links, Verschieben nach rechts
<=, >=, <, >, ==, !=	Vergleich
&	Bitweises AND
^	Bitweises exklusives OR
	Bitweises OR
&&	Logisches AND
	Logisches OR
=, *=, /=, &=, +=, -=, <<=, > >=, &=, ^=,  =	Zuweisung

Viele arithmetische Operatoren wie `*`, `&`, `<` und `>` haben in der Korn- bzw. POSIX-Shell eine Sonderbedeutung. Diese Zeichen müssen in Anführungszeichen gesetzt werden. Wenn Sie beispielsweise den aktuellen Wert von `y` mit 5 multiplizieren und das Ergebnis `y` als neuen Wert verwenden möchten, verwenden Sie den folgenden Ausdruck:

```
let "y = y * 5"
```

Durch die Anführungszeichen wird die Sonderbedeutung des Zeichens \* (Stern) aufgehoben.

Innerhalb von Ausdrücken mit dem Befehl **let** können Operationen gruppiert werden. Im Ausdruck `let "z = q * (z - 10)"`

wird beispielsweise *q* mit dem Ergebnis der Subtraktion von *z* - 10 multipliziert.

Die Korn- bzw. POSIX-Shell unterstützt ein alternatives Format für den Befehl **let**, wenn nur ein einzelner Ausdruck berechnet werden soll. Die Shell behandelt Befehle, die in doppelten runden Klammern, also `(( ))`, stehen, wie in Anführungszeichen gesetzte Ausdrücke. Demzufolge ist der Ausdruck: `((x = x / 3))`

äquivalent zu:

```
let "x = x / 3"
```

Auf benannte Parameter wird innerhalb eines arithmetischen Ausdrucks mit dem Namen verwiesen, ohne dass die Syntax für Parametersubstitution verwendet wird. Wenn auf einen benannten Parameter verwiesen wird, wird sein Wert als arithmetischer Ausdruck ausgewertet.

Eine interne Ganzzahldarstellung eines benannten Parameters wird mit dem Flag **-i** des integrierten Sonderbefehls **typeset** angegeben. Mit dem Flag **-i** wird die arithmetische Berechnung für den Wert jeder Zuordnung eines benannten Parameters ausgeführt. Wird keine arithmetische Basis angegeben, wird sie durch die erste Zuordnung des Parameters festgelegt. Diese Basis wird bei der Parametersubstitution verwendet.

### Zugehörige Konzepte:

„Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 264

Die Korn-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlorientierte Programmiersprache. Sie entspricht dem internationalen Standard für Betriebssysteme POSIX (Portable Operating System Interface for Computer Environment).

„Parameter in der Korn-Shell“ auf Seite 239

Im Folgenden finden Sie eine Beschreibung der Korn-Shellparameter.

### Feldtrennung in der Korn- bzw. POSIX-Shell:

Nach der Durchführung der Befehlssubstitution sucht die Korn-Shell in den Substitutionsergebnissen nach den in der Variablen **IFS** (Internal Field Separator) gefundenen Feldtrennzeichen. Werden Feldtrennzeichen gefunden, teilt die Shell die Substitutionen in separate Argumente.

Explizite Nullargumente (`"` oder `'`) werden beibehalten, und implizite Nullargumente (die sich aus Parametern ohne Werte ergeben) werden verworfen.

- Wenn der Wert von **IFS** ein Leerzeichen, Tabulatorzeichen oder Zeilenvorschubzeichen oder nicht definiert ist, werden alle Folgen von Leerzeichen, Tabulatorzeichen und Zeilenvorschubzeichen am Anfang oder Ende der Eingabe ignoriert. Innerhalb der Eingabe begrenzt eine beliebige Folge dieser Zeichen ein Feld. Die folgende Eingabe trennt beispielsweise die Felder **school** und **days**:  
`<Zeilenumbruch><Leerzeichen><Tabulatorzeichen>school<Tabulatorzeichen><Tabulatorzeichen>days<Leerzeichen>`
- Wenn der Wert von **IFS** ungleich null ist, gelten die folgenden Regeln in der angegebenen Reihenfolge. Die Angabe **IFS Leerzeichen** bezieht sich auf alle Folgen (null oder mehr Exemplare) von Leerzeichen im Wert der Variablen **IFS**. Beispiel: Enthält die Variable **IFS** Leerzeichen/Komma/Tabulatorzeichen, wird jede Folge von Leerzeichen und Tabulatorzeichen als **IFS**-Leerzeichen betrachtet.
  1. **IFS**-Leerzeichen am Anfang und Ende der Eingabe werden ignoriert.
  2. Jedes **IFS**-Zeichen in der Eingabe, bei dem es sich nicht um ein **IFS**-Leerzeichen handelt, dient zusammen mit einem darauffolgenden **IFS**-Leerzeichen als Begrenzer für ein Feld.
  3. Durch **IFS**-Leerzeichen mit einer Länge ungleich null wird ein Feld begrenzt.

## Liste der integrierten Sonderbefehle in der Korn- bzw. POSIX-Shell:

Sonderbefehle sind in die Korn- bzw. POSIX-Shell integriert und werden im Shellprozess ausgeführt.

Eintrag	Beschreibung
<b>:</b> (Doppelpunkt)	Erweitert nur Argumente.
<b>.</b> (Punkt)	Liest eine angegebene Datei und führt anschließend die Befehle aus.
<b>break</b>	Führt mit der Ausführung nach dem nächsten Ende einer <b>for</b> -, <b>while</b> -, <b>until</b> oder <b>select</b> -Schleife fort.
<b>continue</b>	Nimmt die nächste Iteration durch die Befehlsschleife <b>for</b> , <b>while</b> , <b>until</b> oder <b>select</b> wieder auf.
<b>eval</b>	Liest die Argumente als Eingabe für die Shell und führt die entsprechenden Befehle aus.
<b>exec</b>	Führt an Stelle der Shell den im Parameter <i>Argument</i> angegebenen Befehl aus, ohne einen neuen Prozess zu erstellen.
<b>exit</b>	Verlässt die Shell mit dem im Parameter <i>n</i> angegebenen Exit-Status.
<b>export</b>	Markiert Namen für den automatischen Export in die Umgebung von nachfolgend ausgeführten Befehlen.
<b>newgrp</b>	Äquivalent zum Befehl <code>exec/usr/bin/newgrp [Gruppe ...]</code> .
<b>readonly</b>	Markiert die angegebenen Namen als schreibgeschützt.
<b>return</b>	Bewirkt, dass eine Shell zum aufrufenden Script zurückkehrt.
<b>set</b>	Schreibt die Namen und Werte aller Shellvariablen in der Sortierfolge der aktuellen Sprache, wenn keine Optionen oder Argumente angegeben werden.
<b>shift</b>	Benennt Positionsparameter um.
<b>times</b>	Gibt die aufgelaufenen Benutzer- und Systemzeiten für die Shell und die von der Shell ausgeführten Prozesse an.
<b>trap</b>	Führt einen angegebenen Befehl aus, wenn die Shell ein angegebenes oder mehrere angegebene Signal(e) empfängt.
<b>typeset</b>	Setzt Attribute und Werte für Shellparameter.
<b>unset</b>	Inaktiviert die Werte und Attribute der angegebenen Parameter.

## Zugehörige Konzepte:

„Integrierte Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 246

Sonderbefehle sind in die Korn- bzw. POSIX-Shell integriert und werden im Shellprozess ausgeführt.

## Integrierte reguläre Befehle in der Korn- bzw. POSIX-Shell:

Im Folgenden finden Sie eine Liste der integrierten regulären Korn- bzw. POSIX-Shellbefehle.

Eintrag	Beschreibung
<b>alias</b>	Gibt eine Liste der Aliasnamen in der Standardausgabe aus.
<b>bg</b>	Setzt die Ausführung des angegebenen Jobs im Hintergrund fort.
<b>cd</b>	Wechselt vom aktuellen Verzeichnis in das angegebene Verzeichnis oder ersetzt die aktuelle Zeichenfolge durch die angegebene Zeichenfolge.
<b>echo</b>	Schreibt Zeichenfolgen in die Standardausgabe.
<b>fc</b>	Wählt einen Befehlsbereich aus der Ergebnisliste des letzten Befehls <i>HISTSIZE</i> , der am Terminal eingegeben wurde, aus. Führt den angegebenen Befehl erneut aus, nachdem eine Alt-zu-Neu-Substitution ausgeführt wurde.
<b>fg</b>	Setzt die Ausführung des angegebenen Jobs im Vordergrund fort.
<b>getopts</b>	Überprüft den Parameter <i>Argument</i> auf gültige Optionen.
<b>jobs</b>	Listet die Informationen für den angegebenen Job auf.
<b>kill</b>	Sendet das Beendigungssignal <b>TERM</b> an die angegebenen Jobs oder Prozesse.
<b>let</b>	Wertet die angegebenen arithmetischen Ausdrücke aus.
<b>print</b>	Gibt die Shellausgabe aus.
<b>pwd</b>	Äquivalent zum Befehl <code>print -r \$PWD</code> .
<b>read</b>	Liest die Shellvorgabe.
<b>ulimit</b>	Definiert und zeigt die Ressourcengrenzen für Benutzerprozesse an, die in der Datei <code>/etc/security/limits</code> definiert sind.
<b>umask</b>	Legt die Dateiberechtigungen fest.
<b>unalias</b>	Löscht die Parameter aus der Namensliste der Aliasliste.
<b>wait</b>	Wartet auf den angegebenen Job und wird beendet.
<b>whence</b>	Zeigt für jeden angegebenen Namen an, wie die Shell ihn als Befehlsnamen interpretieren würde.

Nähere Informationen hierzu finden Sie im Abschnitt „Integrierte Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 246.

### Zugehörige Konzepte:

„Integrierte Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 246

Sonderbefehle sind in die Korn- bzw. POSIX-Shell integriert und werden im Shellprozess ausgeführt.

### Bedingungsausdrücke in der Korn- bzw. POSIX-Shell:

Ein Bedingungsausdruck wird mit dem zusammengesetzten Befehl `[[` (doppelte eckige Klammern) verwendet, um Dateiattribute zu prüfen und Zeichenfolgen zu vergleichen.

Bei Wörtern zwischen doppelten eckigen Klammern (`[[` und `]]`) erfolgt keine Trennung oder Dateinamensubstitution. Jeder Ausdruck wird aus einem oder mehreren der folgenden unären oder binären Ausdrücke gebildet:

Eintrag	Beschreibung
<b>-a</b> <i>Datei</i>	Wahr, wenn die angegebene Datei eine symbolische Verbindung ist, die auf eine andere existierende Datei zeigt.
<b>-b</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und eine blockorientierte Gerätedatei ist.
<b>-c</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und eine zeichenorientierte Gerätedatei ist.
<b>-d</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und ein Verzeichnis ist.
<b>-e</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden ist.
<b>-f</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und eine gewöhnliche Datei ist.
<b>-g</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und das Definitionsbit für Gruppen-ID ( <b>setgid</b> ) gesetzt ist.
<b>-h</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und eine symbolische Verbindung ist.
<b>-k</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und das Sticky-Bit gesetzt ist.
<b>-n</b> <i>Zeichenfolge</i>	Wahr, wenn die Länge der angegebenen Zeichenfolge ungleich null ist.
<b>-o</b> <i>Option</i>	Wahr, wenn die angegebene Option aktiv ist.
<b>-p</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und eine FIFO-Gerätedatei oder eine Befehlskette ist.
<b>-r</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden ist und vom aktuellen Prozess gelesen werden kann.
<b>-s</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und größer als null ist.
<b>-t</b> <i>Dateideskriptor</i>	Wahr, wenn die angegebene Dateideskriptornummer geöffnet und einem Terminal zugeordnet ist.
<b>-u</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und das Definitionsbit für Benutzer-ID ( <b>setuid</b> ) gesetzt ist.
<b>-w</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und das Schreibbit gesetzt ist. In einem Dateisystem, für das ausschließlich Leseberechtigungen vergeben wurden, kann jedoch nicht in die Datei geschrieben werden, selbst wenn dieser Test eine wahre Bedingung angibt.
<b>-x</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und das Flag <b>execute</b> aktiviert ist. Wenn die angegebene Datei vorhanden und ein Verzeichnis ist, ist der Prozess berechtigt, das Verzeichnis zu durchsuchen.
<b>-z</b> <i>Zeichenfolge</i>	Wahr, wenn die Länge der angegebenen Zeichenfolge gleich null ist.
<b>-L</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und eine symbolische Verbindung ist.
<b>-O</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden ist und der Eigner der Benutzer-ID dieses Prozesses entspricht.
<b>-G</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden ist und ihre Gruppe mit der aktuellen Gruppen-ID dieses Prozesses übereinstimmt.
<b>-S</b> <i>Datei</i>	Wahr, wenn die angegebene Datei vorhanden und eine Socket-Datei ist.
<i>Datei1</i> <b>-nt</b> <i>Datei2</i>	Wahr, wenn <i>Datei1</i> vorhanden und neuer als <i>Datei2</i> ist.
<i>Datei1</i> <b>-ot</b> <i>Datei2</i>	Wahr, wenn <i>Datei1</i> vorhanden und älter als <i>Datei2</i> ist.
<i>Datei1</i> <b>-ef</b> <i>Datei2</i>	Wahr, wenn <i>Datei1</i> und <i>Datei2</i> vorhanden sind und sich auf dieselbe Datei beziehen.
<i>Zeichenfolge1</i> = <i>Zeichenfolge2</i>	Wahr, wenn <i>Zeichenfolge1</i> mit <i>Zeichenfolge2</i> identisch ist.
<i>Zeichenfolge1</i> != <i>Zeichenfolge2</i>	Wahr, wenn <i>Zeichenfolge1</i> nicht mit <i>Zeichenfolge2</i> identisch ist.
<i>Zeichenfolge</i> = <i>Muster</i>	Wahr, wenn die angegebene Zeichenfolge mit dem angegebenen Muster übereinstimmt.
<i>Zeichenfolge</i> != <i>Muster</i>	Wahr, wenn die angegebene Zeichenfolge nicht mit dem angegebenen Muster übereinstimmt.
<i>Zeichenfolge1</i> < <i>Zeichenfolge2</i>	Wahr, wenn <i>Zeichenfolge1</i> bezüglich der ASCII-Werte ihrer Zeichen vor <i>Zeichenfolge2</i> steht.
<i>Zeichenfolge1</i> > <i>Zeichenfolge2</i>	Wahr, wenn <i>Zeichenfolge1</i> bezüglich der ASCII-Werte ihrer Zeichen hinter <i>Zeichenfolge2</i> steht.
<i>Ausdruck1</i> <b>-eq</b> <i>Ausdruck2</i>	Wahr, wenn <i>Ausdruck1</i> mit <i>Ausdruck2</i> identisch ist.

Eintrag	Beschreibung
<i>Ausdruck1</i> -ne <i>Ausdruck2</i>	Wahr, wenn <i>Ausdruck1</i> nicht mit <i>Ausdruck2</i> identisch ist.
<i>Ausdruck1</i> -lt <i>Ausdruck2</i>	Wahr, wenn <i>Ausdruck1</i> kleiner als <i>Ausdruck2</i> ist.
<i>Ausdruck1</i> -gt <i>Ausdruck2</i>	Wahr, wenn <i>Ausdruck1</i> größer als <i>Ausdruck2</i> ist.
<i>Ausdruck1</i> -le <i>Ausdruck2</i>	Wahr, wenn <i>Ausdruck1</i> kleiner-gleich <i>Ausdruck2</i> ist.
<i>Ausdruck1</i> -ge <i>Ausdruck2</i>	Wahr, wenn <i>Ausdruck1</i> größer-gleich <i>Ausdruck2</i> ist.

**Anmerkung:** Bei allen oben genannten Ausdrücken wird der Test für die geöffnete Datei mit der Deskriptornummer *n* ausgeführt, wenn die Variable *Datei* dem Ausdruck */dev/fd/n* entspricht. *n* steht für eine ganze Zahl.

Aus diesen Basiselementen oder kleineren Teilen können zusammengesetzte Ausdrücke gebildet werden, indem einer der folgenden Ausdrücke verwendet wird. Die Ausdrücke sind nach absteigender Rangfolge aufgeführt:

Eintrag	Beschreibung
( <i>Ausdruck</i> )	Wahr, wenn der angegebene Ausdruck wahr ist. Wird verwendet, um Ausdrücke zu gruppieren.
! <i>Ausdruck</i>	Wahr, wenn der angegebene Ausdruck falsch ist.
<i>Ausdruck1</i> && <i>Ausdruck2</i>	Wahr, wenn <i>Ausdruck1</i> und <i>Ausdruck2</i> wahr sind.
<i>Ausdruck1</i>    <i>Ausdruck2</i>	Wahr, wenn entweder <i>Ausdruck1</i> oder <i>Ausdruck2</i> wahr ist.

### Zeichen in der Korn- bzw. POSIX-Shell in Anführungszeichen setzen:

Wenn die Korn- bzw. POSIX-Shell ein Zeichen als reguläres Zeichen und nicht in der dem Zeichen normalerweise zugeordneten Sonderbedeutung liest, muss das Zeichen in *Anführungszeichen* gesetzt werden.

Jedes Metazeichen hat in der Shell eine Sonderbedeutung und wirkt als Begrenzer für ein Wort, wenn es nicht in Anführungszeichen gesetzt ist. Die folgenden Zeichen werden von der Korn- bzw. POSIX-Shell als Metazeichen interpretiert und müssen in Anführungszeichen gesetzt werden, um in ihrer normalen Bedeutung verwendet zu werden.

- Pipe (|)
- Et-Zeichen (&)
- Semikolon (;)
- Kleiner-als-Zeichen (<) und Größer-als-Zeichen (>)
- linke runde Klammer (()) und rechte runde Klammer ())
- Dollarzeichen (\$)
- umgekehrtes Anführungszeichen (`) und einfaches Anführungszeichen (')
- Backslash (\)
- doppeltes Anführungszeichen (")
- Zeilenvorschubzeichen
- Leerzeichen
- Tabulatorzeichen

Um die Sonderbedeutung eines Metazeichens außer Kraft zu setzen, muss eine der Methoden für das Anführen von Angaben aus der nachfolgend aufgeführten Liste verwendet werden.



Eintrag	Beschreibung
<b>Backslash</b>	Bei Verwendung eines Backslash (\), der nicht in Anführungszeichen gesetzt ist, wird der Literalwert des nachfolgenden Zeichens, außer im Falle eines Zeilenvorschubzeichens, beibehalten. Ist nach dem Backslash ein Zeilenvorschubzeichen angegeben, interpretiert die Shell dieses als Zeilenfortsetzung.
<b>Einfache Anführungszeichen</b>	<p>Werden Zeichen in einfache Anführungszeichen eingeschlossen ( ' '), behalten diese ihren Literalwert. Ein einfaches Anführungszeichen darf nicht in einfache Anführungszeichen gesetzt werden.</p> <p>Auch mithilfe eines Backslash kann ein einfaches Anführungszeichen nicht innerhalb einer Zeichenfolge, die in einfache Anführungszeichen eingeschlossen ist, stehen. Ein eingebettetes Anführungszeichen kann z. B. durch folgende Eingabe erstellt werden: 'a'\ 'b'. Dies ergibt als Ausgabe a'b.</p>
<b>Doppelte Anführungszeichen</b>	<p>Wenn Sie Zeichen in doppelte Anführungszeichen ( " ") einschließen, wird der Literalwert aller Zeichen innerhalb der doppelten Anführungszeichen mit Ausnahme des Dollarzeichens, des umgekehrten Anführungszeichens und des Backslash, wie folgt beibehalten:</p> <p><b>\$</b> Das Dollarzeichen behält seine Sonderbedeutung bei. Es dient der Einleitung einer Parametererweiterung, als Befehlssubstitution und als arithmetische Erweiterung.</p> <p>Die Eingabezeichen innerhalb der Zeichenfolge in Anführungszeichen, die wiederum in \$( und ) eingeschlossen sind, sind von den doppelten Anführungszeichen nicht betroffen, sondern definieren den Befehl, dessen Ausgabe \$(...) ersetzt, wenn das Wort erweitert wird.</p> <p>Die Zeichenfolge zwischen \${ und } muss eine gerade Anzahl doppelter oder einfacher Anführungszeichen ohne Escape-Zeichen enthalten, sofern welche verwendet werden. Ein vorangestellter umgekehrter Schrägstrich muss als Abbruchzeichen für ein Literal { oder } verwendet werden.</p> <p><b>`</b> Die Sonderbedeutung des umgekehrten Anführungszeichens wird beibehalten und stellt eine andere Form der Befehlssubstitution dar. Der Teil der Zeichenfolge in Anführungszeichen, der mit dem ersten umgekehrten Anführungszeichen beginnt und mit dem nächsten umgekehrten Anführungszeichen endet und dem kein Backslash vorangestellt ist, definiert den Befehl, dessen Ausgabe `...` ersetzt, wenn das Wort erweitert wird.</p> <p><b>\</b> Der Backslash behält seine Sonderbedeutung als Abbruchzeichen nur dann bei, wenn eines der folgenden Zeichen hinter dem Backslash angegeben ist: \$, `, ", \ oder Zeilenvorschubzeichen.</p>

Vor einem doppelten Anführungszeichen muss ein Backslash angegeben werden, um dieses in doppelte Anführungszeichen einzuschließen. Werden doppelte Anführungszeichen verwendet und ist unmittelbar nach einem Backslash ein Zeichen angegeben, das eine Sonderbedeutung hat, wird der Backslash gelöscht und das darauffolgende Zeichen wörtlich (als Literal) interpretiert. Ist nach dem Backslash ein Zeichen ohne Sonderbedeutung angegeben, bleiben der Backslash und das unmittelbar darauffolgende Zeichen unverändert. Beispiel:

```
"\$" -> $
"\a" -> \a
```

Die folgenden Bedingungen gelten für Meta- und Anführungszeichen in der Korn- bzw. POSIX-Shell:

- Die Bedeutung von \$\* (Dollarzeichen, Stern) und @\$ (Dollarzeichen, At-Zeichen) sind bei der Verwendung als Parameterzuordnungswert oder Dateiname identisch, wenn sie nicht in Anführungszeichen gesetzt sind.
- Wenn "\$\*" (doppelte Anführungszeichen, Dollarzeichen, Stern, doppelte Anführungszeichen) als Befehlsargument verwendet wird, entspricht dies "\$1\$d\$2d...", wobei d das erste Zeichen des IFS-Parameters ist.
- "\$@" (doppelte Anführungszeichen, At-Zeichen, Stern, doppelte Anführungszeichen) ist identisch mit "\$1" "\$2" ....

- Innerhalb von umgekehrten Anführungszeichen (``) werden durch den Backslash die Zeichen \ (Backslash), ' (einfaches Anführungszeichen) und \$ (Dollarzeichen) in Anführungszeichen gesetzt. Stehen die umgekehrten Anführungszeichen innerhalb von doppelten Anführungszeichen (" "), werden durch den Backslash auch doppelte Anführungszeichen in Anführungszeichen gesetzt.
- Die Parameter- und Befehlssubstitution erfolgt innerhalb der doppelten Anführungszeichen (" ").
- Die Sonderbedeutung von reservierten Wörtern oder Aliasnamen wird aufgehoben, indem ein beliebiges Zeichen des reservierten Worts in Anführungszeichen gesetzt wird. Funktionsnamen oder Namen von integrierten Befehlen können nicht in Anführungszeichen gesetzt werden.

### Restricted Korn Shell:

Mit der Restricted Korn Shell werden Anmeldenamen und Ausführungsumgebungen eingerichtet, deren Leistungsspektrum geringer ist als das der regulären Korn-Shell.

Die Befehle **rksh** und **ksh -r** öffnen die Restricted Korn Shell. Die Aktionen dieser Befehle entsprechen im Wesentlichen denen des Befehls **ksh**. Eine Ausnahme bilden die folgenden Aktionen, die in der Restricted Shell nicht zulässig sind:

- aktuelles Arbeitsverzeichnis wechseln
- Wert der Variablen *SHELL*, *ENV* und *PATH* setzen
- Pfadnamen eines Befehls mit einem Schrägstrich / angeben
- Ausgabe eines Befehls mit > (Rechtspfeil), >| (Rechtspfeil, Pipe-Symbol), <> (Linkspfeil, Rechtspfeil) oder >> (zwei Rechtspfeile) umleiten.

Stellt die Restricted Korn-Shell fest, dass es sich bei einem auszuführenden Befehl um eine Shellprozedur handelt, ruft sie zur Ausführung des Befehls die Korn-Shell auf. Auf diese Weise ist es möglich, dem Endbenutzer über ein eingeschränktes Befehlsmenü Shellprozeduren zur Verfügung zu stellen, die auf das gesamte Leistungsspektrum der Korn-Shell zugreifen. In dieser Situation wird davon ausgegangen, dass der Benutzer nicht über Schreib- und Ausführungsberechtigungen in demselben Verzeichnis verfügt.

Wird der Parameter **Datei** [*Parameter*] beim Aufruf der Korn-Shell angegeben, führt die Shell die im Parameter **Datei** angegebene Scriptdatei mit allen angegebenen Parametern aus. Die angegebene Scriptdatei muss über Lesezugriff verfügen. Die **setuid**- und **setgid**-Einstellungen für Scriptdateien werden ignoriert. Anschließend liest die Shell die Befehle. Geben Sie keine Scriptdatei an, wenn Sie das Flag **-c** oder **-s** verwenden.

Wird die Shell mit dem Befehl **rksh** gestartet, führt die Shell nach der Interpretation der Dateien *.profile* und */etc/environment* Einschränkungen ein. Deshalb kann der Autor der Datei *.profile* Benutzeraktionen genau steuern, indem er Konfigurationsaktionen ausführt und den Benutzer in ein bestimmtes Verzeichnis (möglicherweise nicht das Anmeldeverzeichnis) stellt. Ein Administrator kann durch entsprechende Änderung der Variablen *PATH* ein Verzeichnis mit Befehlen im Verzeichnis */usr/rbin* einrichten, das vom Befehl **rksh** verwendet werden kann. Wird die Shell mit dem Befehl **ksh -r** aufgerufen, führt sie die Einschränkungen bei der Interpretation der Dateien *.profile* ein.

Wird die Restricted Korn Shell mit dem Befehl **rksh** aufgerufen, liest sie die Datei *.profile* des Benutzers (*\$HOME/.profile*). Sie verhält sich dabei wie eine reguläre Korn-Shell, mit der Ausnahme, dass die Shell durch eine Unterbrechung sofort verlassen und keine Rückkehr auf Befehlsebene erfolgt.

### Reservierte Wörter in der Korn- bzw. POSIX-Shell:

Die folgenden reservierten Wörter haben für die Korn- bzw. POSIX-Shell eine Sonderbedeutung.

```
! case do
done elif else
esac fi for
```

```
function if in
select then time
until while {
} [[]]
```

Die reservierten Wörter werden nur erkannt, wenn sie ohne Anführungszeichen angegeben und an einer der folgenden Positionen verwendet werden:

- als erstes Wort eines Befehls,
- als erstes Wort, das einem der reservierten Wörter außer **case**, **for** oder **in** folgt,
- als drittes Wort in einem Befehl **case** oder **for** (nur **in** ist in diesem Fall gültig).

### Erweiterte Korn-Shell (ksh93):

Zusätzlich zur Standard-Korn-Shell des Systems (/usr/bin/ksh) stellt AIX eine erweiterte Version der Korn-Shell unter /usr/bin/ksh93 zur Verfügung. Diese erweiterte Version ist zum größten Teil mit der aktuellen Standardversion kompatibel und enthält neue zusätzliche Funktionen, die in /usr/bin/ksh nicht verfügbar sind.

Einige Scripts werden in der Korn-Shell ksh93 anders ausgeführt als in der Standardshell, weil die Variablenverarbeitung sich in den beiden Shells unterscheidet.

**Anmerkung:** Außerdem ist eine eingeschränkte Version der erweiterten Korn-Shell mit dem Namen rksh93 verfügbar.

Die folgenden Funktionen sind im Gegensatz zur Korn-Shell /usr/bin/ksh93 in der Korn-Shell /usr/bin/ksh nicht verfügbar:

Eintrag	Beschreibung
<b>Arithmetische Erweiterungen</b>	Sie können die <b>libm</b> -Funktionen (mathematische Funktionen, die typisch für die Programmiersprache C sind) in arithmetischen Ausdrücken wie <code>\$ value=\$((sqrt(9))</code> verwenden. Es sind weitere arithmetische Operatoren verfügbar, darunter das unäre <code>+</code> , <code>++</code> , <code>--</code> und das Konstrukt <code>?:</code> (z. B. <code>"x ? y : z"</code> ) sowie der Operator <code>,</code> (Komma). Die Zahlendarstellung wird bis zur Basis 64 unterstützt. Gleitkommaberechnungen werden ebenfalls unterstützt. Mit dem Befehl <b>"typeset -E"</b> (exponentiell) kann die Anzahl der signifikanten Ziffern und mit dem Befehl <b>"typeset -F"</b> (Gleitkomma) die Anzahl der Dezimalstellen für arithmetische Variablen angegeben werden. Die Variable <b>SECONDS</b> gibt die Werte jetzt bis auf Hundertstelsekunden genau und nicht mehr nur sekundengenau an.
<b>Verbundvariablen</b>	Verbundvariablen werden unterstützt. Mit einer Verbundvariablen können Sie mehrere Werte unter einem Variablennamen angeben. Die Werte werden jeweils einer untergeordneten Variablen zugeordnet, die durch einen Punkt ( <code>.</code> ) von der übergeordneten Variablen getrennt wird. Beispiel: <pre>\$ meinevar=( x=1 y=2 ) \$ print "\${meinevar.x}" 1</pre>
<b>Zuweisung von Verbundvariablen</b>	Zuweisungen von Verbundvariablen werden bei der Initialisierung von Arrays (indexierte und assoziative Arrays) unterstützt. Die Zuordnungswerte werden in runde Klammern gesetzt. Beispiel: <pre>\$ numbers=( zero one two three ) \$ print \${numbers[0]} \${numbers[3]} zero three</pre>
<b>Assoziative Arrays</b>	Ein assoziatives Array ist ein Array mit einer Zeichenfolge als Index.  Wenn Sie den Befehl <b>typeset</b> mit dem Flag <b>-A</b> angeben, können Sie in der ksh93 assoziative Arrays festlegen. Beispiel: <pre>\$ typeset -A teammates \$ teammates=( [john]=smith [mary]=jones ) \$ print \${teammates[mary]} jones</pre>

Eintrag	Beschreibung
<b>Verweise auf Variablenamen</b>	<p>Wenn Sie den Befehl <b>typeset</b> mit dem Flag <b>-n</b> angeben, können Sie einen Variablenamen als Verweis auf einen anderen Variablenamen zuordnen. Wenn der Wert einer Variablen geändert wird, ändert sich auf diese Weise auch der Wert der Variablen, auf die verwiesen wird. Beispiel:</p> <pre>\$ greeting="hello" \$ typeset -n welcome=greeting    # erstellt die Referenz \$ welcome="hi there"             # überschreibt den vorherigen Wert \$ print \$greeting hi there</pre>
<b>Parametererweiterungen</b>	<p>Die folgenden Konstrukte für Parametererweiterungen sind verfügbar:</p> <ul style="list-style-type: none"> <li>• <code>\${!varname}</code> ist der Name der Variablen.</li> <li>• <code>\${!varname[@]}</code> gibt die Indizes für das Array <code>varname</code> an.</li> <li>• <code>\${param:offset}</code> ist eine Unterzeichenfolge von <code>param</code>, die bei <code>offset</code> beginnt.</li> <li>• <code>\${param:offset:num}</code> ist eine Unterzeichenfolge von <code>param</code>, die bei <code>offset</code> beginnt und bis zur mit <code>num</code> angegebenen Anzahl von Zeichen reicht.</li> <li>• <code>\${@:offset}</code> gibt alle positionsgebundenen Parameter ab <code>offset</code> an.</li> <li>• <code>\${@:offset:num}</code> gibt die Anzahl (<code>num</code>) positionsgebundener Parameter ab dem angegebenen <code>Offset</code> an.</li> <li>• <code>\${param/Muster/repl}</code> ergibt <code>param</code>, wobei das erste Vorkommen des <code>Musters</code> durch <code>repl</code> ersetzt wird.</li> <li>• <code>\${param//Muster/repl}</code> ergibt <code>param</code>, wobei jedes Vorkommen des <code>Musters</code> durch <code>repl</code> ersetzt wird.</li> <li>• <code>\${param/#Muster/repl}</code> Wenn <code>param</code> mit <code>Muster</code> beginnt, wird <code>param</code> durch <code>repl</code> ersetzt.</li> <li>• <code>\${param/%Muster/repl}</code> Wenn <code>param</code> mit <code>Muster</code> endet, wird <code>param</code> durch <code>repl</code> ersetzt.</li> </ul>
<b>Discipline-Funktionen</b>	<p>Eine Discipline-Funktion ist eine Funktion, die einer bestimmten Variable zugeordnet ist. Auf diese Weise können Sie eine Funktion definieren und jedes Mal aufrufen, wenn auf diese Variable verwiesen wird, unabhängig davon, ob sie gesetzt ist. Die Funktionen haben das Format <code>varname.function</code>, wobei <code>varname</code> für den Namen der Variablen und <code>function</code> für die Discipline-Funktion steht. Die vordefinierten Discipline-Funktionen sind <b>get</b>, <b>set</b> und <b>unset</b>.</p> <ul style="list-style-type: none"> <li>• Die Funktion <code>varname.get</code> wird jedes Mal aufgerufen, wenn auf <code>varname</code> verwiesen wird. Wenn die Sondervariable <code>.sh.value</code> in dieser Funktion gesetzt ist, nimmt <code>varname</code> diesen Wert an. Ein einfaches Beispiel ist die Uhrzeit: <pre>\$ function time.get &gt; { &gt;     .sh.value=\$(date +%r) &gt; } \$ print \$time 09:15:58 AM \$ print \$time    # it will change in a few seconds 09:16:04 AM</pre> </li> <li>• Die Funktion <code>varname.set</code> wird jedes Mal aufgerufen, wenn <code>varname</code> gesetzt ist. Die Variable <code>.sh.value</code> nimmt den zugeordneten Wert an. Der <code>varname</code> zugeordnete Wert ist der Wert, den <code>.sh.value</code> nach Abschluss der Funktion annimmt. Beispiel: <pre>\$ function adder.set &gt; { &gt;     let .sh.value=" \$ { .sh.value } + 1" &gt; } \$ adder=0 \$ echo \$adder 1 \$ adder=\$adder \$ echo \$adder 2</pre> </li> <li>• Die Funktion <code>varname.unset</code> wird jedes Mal ausgeführt, wenn die Definition von <code>varname</code> aufgehoben wird. Die Definition der Variablen wird erst dann aufgehoben, wenn die Definition in der Funktion selbst aufgehoben wird. Andernfalls behält sie ihren Wert.</li> </ul> <p>In allen Discipline-Funktionen wird die Sondervariable <code>.sh.name</code> durch den Namen der Variablen ersetzt, wohingegen <code>.sh.subscript</code> den Wert des Variablensubskripts (sofern vorhanden) annimmt.</p>
<b>Funktionsumgebungen</b>	<p>Im Format <code>function myfunc</code> deklarierte Funktionen werden in einer separaten Funktionsumgebung ausgeführt. Im Format <code>myfunc()</code> deklarierte Funktionen werden in derselben Umgebung wie die übergeordnete Shell ausgeführt.</p>

Eintrag	Beschreibung
<b>Variablen</b>	Variablen, die mit <code>.sh.</code> beginnen, sind von der Shell reserviert und haben eine Sonderbedeutung. In der Beschreibung der Discipline-Funktionen in dieser Tabelle finden Sie eine Erläuterung von <code>.sh.name</code> , <code>.sh.value</code> und <code>.sh.subscript</code> . Außerdem verfügbar ist die Variable <code>.sh.version</code> , die die Version der Shell angibt.
<b>Rückgabewerte von Befehlen</b>	Die Rückgabewerte von Befehlen sind wie folgt: <ul style="list-style-type: none"> <li>• Falls der auszuführende Befehl nicht vorhanden ist, ist der Rückgabewert 127.</li> <li>• Falls der auszuführende Befehl zwar vorhanden, aber nicht ausführbar ist, ist der Rückgabewert 126.</li> <li>• Falls der Befehl zwar ausgeführt, aber dann durch ein Signal abgebrochen wird, setzt sich der Rückgabewert aus 256 und der Signalnummer zusammen.</li> </ul>
<b>Suchregeln für PATH</b>	Die Reihenfolge, in der eingegebene Befehle aufgelöst werden, ist wie folgt: bestimmte integrierte Befehle, Funktionen (einschließlich der Funktionen in <code>FPATH</code> -Verzeichnissen), andere integrierte Befehle.
<b>Shellprotokoll</b>	Mit dem Befehl <code>hist</code> können Sie das Befehlsprotokoll von Shells anzeigen und ändern. In der ksh-Shell wird dazu der Befehl <code>fc</code> verwendet. Der Befehl <code>fc</code> ist ein Aliasname für <code>hist</code> . Die Variablen sind <code>HISTCMD</code> und <code>HISTEDIT</code> . Die Variable <code>HISTCMD</code> wird nach jedem ausgeführten Befehl im aktuellen Shellprotokoll um eins erhöht. Die Variable <code>HISTEDIT</code> gibt an, welcher Editor für den Befehl <code>hist</code> zu verwenden ist.
<b>Integrierte Befehle</b>	Die erweiterte Korn-Shell enthält die folgenden integrierten Befehle: <ul style="list-style-type: none"> <li>• Der Befehl <code>builtin</code> listet alle verfügbaren integrierten Befehle auf.</li> <li>• Der Befehl <code>printf</code> funktioniert ähnlich wie die C-Bibliotheksroutine <code>printf()</code>. Weitere Informationen finden Sie in der Beschreibung des Befehls <code>printf</code>.</li> <li>• Der Befehl <code>disown</code> verhindert, dass die Shell ein Signal <code>SIGHUP</code> an den angegebenen Befehl sendet.</li> <li>• Der Befehl <code>getconf</code> funktioniert ähnlich wie der eigenständige Befehl <code>/usr/bin/getconf</code>. Ziehen Sie die Beschreibung des Befehls <code>getconf</code> zu Rate.</li> <li>• Mit dem integrierten Befehl <code>read</code> können die folgenden Flags angegeben werden: <ul style="list-style-type: none"> <li>– Mit <code>read -d {Zeichen}</code> können Sie einen Zeichenbegrenzer anstelle des standardmäßig verwendeten Zeilenvorschubzeichens angeben.</li> <li>– Mit <code>read -t {Sekunden}</code> können Sie ein Zeitlimit (in Sekunden) angeben, nach dem eine Zeitlimitüberschreitung für den Befehl <code>read</code> ausgelöst wird. Wenn eine Zeitlimitüberschreitung für den Befehl <code>read</code> ausgelöst wird, gibt er den Wert <code>FALSE</code> zurück.</li> </ul> </li> <li>• Mit dem integrierten Befehl <code>exec</code> können die folgenden Flags angegeben werden: <ul style="list-style-type: none"> <li>– <code>exec -a {Name} {Befehl}</code> gibt an, dass Argument 0 von <code>Befehl</code> durch <code>Name</code> ersetzt wird.</li> <li>– <code>exec -c {Befehl}</code> weist den Befehl <code>exec</code> an, vor der Ausführung von <code>Befehl</code> die Umgebung zu löschen.</li> </ul> </li> <li>• Mit dem integrierten Befehl <code>kill</code> können die folgenden Flags angegeben werden: <ul style="list-style-type: none"> <li>– Mit <code>kill -n {Signalnummer}</code> wird die Nummer eines Signals angegeben, das an einen Prozess gesendet werden soll. Mit <code>kill -s {Signalname}</code> wird ein Signalname angegeben.</li> <li>– Mit <code>kill -l</code> (ohne Argumente) werden alle Signalnamen ohne Nummern aufgelistet.</li> </ul> </li> <li>• Mit dem integrierten Befehl <code>whence</code> können die folgenden Flags angegeben werden: <ul style="list-style-type: none"> <li>– Das Flag <code>-a</code> zeigt alle Übereinstimmungen und nicht nur die erste an.</li> <li>– Das Flag <code>-f</code> weist <code>whence</code> an, nicht nach Funktionen zu suchen.</li> </ul> </li> <li>• Für die Befehle <code>print</code> und <code>echo</code> wird eine Escape-Zeichenfolge verwendet. Die Abbruchtaste (Escape) kann mit die Zeichenfolge <code>\E</code> dargestellt werden.</li> <li>• Alle regulären integrierten Befehle erkennen das Flag <code>-?</code>, das die Syntax für den angegebenen Befehl anzeigt.</li> <li>• Der integrierte Befehl <code>getopts</code> erfordert, dass <code>optstring</code> ein führendes Pluszeichen (+) enthält, damit Optionen zugelassen werden, die mit dem Plussymbol (+) beginnen.</li> </ul>

Eintrag	Beschreibung
<b>Weitere Unterschiede zwischen der Korn-Shell ksh und der Korn-Shell ksh93</b>	<p>Im Folgenden sind weitere Unterschiede zwischen den beiden Versionen der Korn-Shell aufgeführt:</p> <ul style="list-style-type: none"> <li>• In der Korn-Shell ksh93 können Sie mit dem integrierten Befehl <b>typeset -fx</b> keine Funktionen exportieren.</li> <li>• In der Korn-Shell ksh93 können Sie mit dem integrierten Befehl <b>alias -x</b> keinen Aliasnamen exportieren.</li> <li>• In der Korn-Shell ksh93 wird ein Dollarzeichen, gefolgt von einem einfachen Anführungszeichen ('\$') als ANSI-C-Zeichenfolge interpretiert. Sie müssen das Dollarzeichen in Anführungszeichen ("'\$'") setzen, um das alte Verhalten (der ksh) zu erhalten.</li> <li>• Die Logik für die Syntaxanalyse von Argumenten für die integrierten Befehle der Korn-Shell ksh93 wurde geändert. Die nicht dokumentierten Kombinationen für die Syntaxanalyse von Argumenten für die integrierten Befehle der Korn-Shell ksh funktionieren nicht in der Korn-Shell ksh93. Der Befehl <b>typeset -i4</b> funktioniert beispielsweise ähnlich wie der Befehl <b>typeset -i4</b> in der Korn-Shell ksh, funktioniert aber nicht in der Korn-Shell ksh93.</li> <li>• In der Korn-Shell ksh93 werden Befehlssubstitutionen und arithmetische Erweiterungen während der Erweiterung in den Sonderumgebungsvariablen PS1, PS3 und ENV durchgeführt. Deshalb müssen Sie dem Gravissymbol (^) sowie dem Dollarzeichen und der linken runden Klammer (\$) einen Backslash (\) als Escape-Zeichen voranstellen, um das alte Verhalten beizubehalten. Beispielsweise ordnet die Korn-Shell ksh <code>x='\${Name}\tOperator'</code> folgendermaßen zu: <code>\$Name\tOperator</code>. Die Korn-Shell ksh93 hingegen erweitert <code>\t</code> und nimmt die Zuordnung folgendermaßen vor: <code>Name&lt;\t erweitert&gt;Operator</code>. Wenn Sie das Verhalten der Korn-Shell beibehalten möchten, müssen Sie \$ in Anführungszeichen setzen, z. B. <code>x='\${Name}\tOperator'</code>.</li> <li>• Die Variable <code>ERRNO</code> wurde in der Korn-Shell ksh93 entfernt.</li> <li>• In der Korn-Shell ksh93 werden Dateinamen für nicht interaktive Shells nach dem Umleitungssymbol nicht erweitert.</li> <li>• In der Korn-Shell ksh93 müssen Sie die Option <code>-t</code> des Befehls <b>alias</b> verwenden, um Aliasnamen mit absoluter Pfadangabe anzuzeigen. Das Feature für Aliasnamen mit absoluter Pfadangabe ist jetzt veraltet. Deshalb werden die angezeigten Aliasnamen möglicherweise nicht mehr verfolgt.</li> <li>• In der Korn-Shell ksh93 werden mit der Tastenkombination Strg+T im emacs-Modus das aktuelle und das vorherige Zeichen vertauscht. In der Korn-Shell ksh werden mit der Tastenkombination Strg+T das aktuelle und das darauffolgende Zeichen vertauscht.</li> <li>• Die Korn-Shell ksh93 lässt die Verwendung unpaariger runder Klammern in <code>\${Name Operator Wert}</code> nicht zu. Beispielsweise muss in <code>\${Name-() }</code> ein Escape-Zeichen verwendet werden, z. B. <code>\${Name-()\{}</code>, damit der Befehl in beiden Versionen funktioniert.</li> <li>• In der Korn-Shell ksh93 listet der Befehl <b>kill -l</b> nur die Signalnamen, aber nicht die zugehörigen numerischen Werte auf.</li> </ul>

### Exit-Status in der Korn- bzw. POSIX-Shell:

Werden von der Shell Fehler, z. B. Syntaxfehler, festgestellt, gibt die Shell einen Exit-Status ungleich null zurück. Andernfalls gibt die Shell den Exit-Status des zuletzt ausgeführten Befehls zurück.

Die Shell meldet Fehler bei der Ausführung, indem der Befehls- oder Funktionsname und die entsprechende Fehlerbedingung ausgegeben wird. Ist die Nummer der Zeile, in der der Fehler festgestellt wurde, größer als 1, wird zusätzlich die Zeilennummer in eckigen Klammern ([ ]) nach dem Befehls- oder Funktionsnamen ausgegeben.

Wenn in einem integrierten Sonderbefehl oder einer andere Art von Befehl in einer nicht interaktiven Shell ein Fehler auftritt, schreibt die Shell eine Diagnosenachricht (siehe folgende Tabelle):

Fehler	Integrierter Sonderbefehl	Andere Dienstprogramme
Shell-Syntaxfehler	wird beendet	wird beendet
Dienstprogrammssyntaxfehler (Fehler bei Option oder Operand)	wird beendet	wird nicht beendet
Umleitungsfehler	wird beendet	wird nicht beendet
Variablenzuordnungsfehler	wird beendet	wird nicht beendet
Erweiterungsfehler	wird beendet	wird beendet
Befehl nicht gefunden	nicht anwendbar	wird evtl. beendet
Im aktuellen Shellkontext auszuführendes Script nicht gefunden	wird beendet	nicht anwendbar

Falls einer der Fehler mit der Angabe "wird evtl. beendet" in einer Subshell auftritt, wird die Subshell (evtl.) mit einem Status ungleich null beendet. Das Script, in dem die Subshell enthalten ist, wird aufgrund dieses Fehlers jedoch nicht beendet.

In allen in der Tabelle aufgeführten Fällen schreibt eine interaktive Shell eine Diagnosenachricht in die Standardfehlerausgabe und wird nicht beendet.

### Parameter in der Korn-Shell:

Im Folgenden finden Sie eine Beschreibung der Korn-Shellparameter.

Ein Parameter ist wie folgt definiert:

- Kennung für eines der folgenden Zeichen: Stern (\*), At-Zeichen (@), Nummernzeichen (#), Fragezeichen (?), Minuszeichen (-), Dollarzeichen (\$) und Ausrufezeichen (!). Diese Parameter werden als *Sonderparameter* bezeichnet.
- Ein Argument, das durch eine Nummer (*Positionsparameter*) bezeichnet wird.
- Ein Parameter, der durch eine ID mit einem Wert und keinem oder mehr Attribut(en) bezeichnet wird (*benannte Parameter/Variablen*).

Mit dem integrierten Sonderbefehl **typeset** werden benannten Parametern Werte und Attribute zugeordnet. Die von der Korn-Shell unterstützten Attribute werden im Abschnitt zum integrierten Sonderbefehl **typeset** beschrieben. Exportierte Parameter übergeben Werte und Attribute an die Umgebung.

Der Wert eines benannten Parameters wird folgendermaßen zugeordnet:

Name=Wert [ Name=Wert ] ...

Wird das Integer-Attribut **-i** für den Parameter **Name** gesetzt, wird der Parameter **Wert** durch arithmetische Berechnung ermittelt.

Die Shell unterstützt eindimensionale Arrays. Auf Elemente eines Array-Parameters wird durch einen Index verwiesen. Ein Index wird durch einen arithmetischen Ausdruck gekennzeichnet, der in eckige Klammern ( [ ] ) eingeschlossen ist. Geben Sie Folgendes ein, um einem Array Werte zuzuordnen: `set -A Name Wert`. Der Wert aller Indizes muss zwischen 0 und 511 liegen. Arrays müssen nicht deklariert werden. Jeder Verweis auf einen benannten Parameter mit einem gültigen Index ist zulässig, und ein Array wird, falls erforderlich, erstellt. Der Verweis auf ein Array ohne Index entspricht einem Verweis auf das Element 0.

Mit dem Sonderbefehl **set** werden positionsgebundenen Parametern Werte zugeordnet. Der Parameter **\$0** wird beim Aufruf der Shell vom Argument 0 gesetzt. Das Dollarzeichen (\$) wird verwendet, um ersetzbare Parameter einzuführen.

### Zugehörige Konzepte:

„Shell starten“ auf Seite 267

Sie können die Korn-Shell mit dem Befehl **ksh**, dem Befehl **psh** (POSIX-Shell) oder dem Befehl **exec** starten.

„Funktionen in der Korn-Shell“ auf Seite 268

Das reservierte Wort **function** definiert Shellfunktionen. Die Shell liest und speichert Funktionen intern. Aliasnamen werden beim Lesen der Funktion aufgelöst. Funktionen werden von der Shell auf dieselbe Weise ausgeführt wie Befehle, wobei die Argumente als positionsgebundene Parameter übergeben werden.

„Arithmetische Berechnungen in der Korn- bzw. POSIX-Shell“ auf Seite 228

Mit dem integrierten Befehl **let** der Korn- bzw. POSIX-Shell können Sie Ganzzahlberechnungen durchführen.

### Zugehörige Verweise:

„Zusammengesetzte Befehle in der Korn-Shell“ auf Seite 265

Ein zusammengesetzter Befehl kann eine Liste einfacher Befehle oder eine Kette sein oder mit einem reservierten Wort beginnen. Am häufigsten werden Sie zusammengesetzte Befehle wie **if**, **while** und **for** verwenden, wenn Sie Shell-Skripts schreiben.

*Parametersubstitution in der Korn- bzw. POSIX-Shell:*

In der Korn- bzw. POSIX-Shell können Parametersubstitutionen durchgeführt werden.

Nachfolgend werden ersetzbare Parameter aufgelistet:

Eintrag	Beschreibung
<code>\${Parameter}</code>	Die Shell liest alle Zeichen vom Dollarzeichen mit linker geschweifter Klammer ( <code>{</code> ) bis zur zugehörigen rechten geschweiften Klammer ( <code>}</code> ) als Teil desselben Wortes, auch wenn das Wort geschweifte Klammern oder Metazeichen enthält. Der Wert des angegebenen Parameters wird, falls vorhanden, ersetzt. Die geschweiften Klammern sind erforderlich, wenn auf den Parameter <i>Parameter</i> ein Buchstabe, eine Ziffer oder ein Unterstreichungszeichen folgt, das nicht als Teil seines Namens interpretiert werden soll, oder wenn ein benannter Parameter indiziert ist.  Enthält der angegebene Parameter eine oder mehrere Ziffer(n), handelt es sich um einen <i>positionsgebundenen Parameter</i> . Positionsgebundene Parameter mit mehr als einer Ziffer müssen in geschweifte Klammern eingeschlossen werden. Ist der Wert der Variablen ein Stern (*) oder ein At-Zeichen (@), wird jeder positionsgebundene Parameter, der mit <code>\$1</code> beginnt, ersetzt (getrennt durch ein Feldtrennzeichen). Wird eine Array-Kennung mit einem Stern * oder einem At-Zeichen (@) als Index verwendet, wird der Wert für jedes Element (getrennt durch ein Feldtrennzeichen) ersetzt.
<code>\${#Parameter}</code>	Ist der Wert des Parameters <i>Parameter</i> ein Stern (*) oder ein At-Zeichen (@), wird die Anzahl der positionsgebundenen Parameter ersetzt. Andernfalls wird die im Parameter <i>Parameter</i> angegebene Feldlänge ersetzt.
<code>\${#ID[*]}</code>	Die Anzahl der Elemente in dem im Parameter <i>ID</i> angegebenen Array wird ersetzt.
<code>\${Parameter:-Wort}</code>	Ist der Parameter <i>Parameter</i> gesetzt und ungleich null, wird dessen Wert ersetzt. Andernfalls wird der Wert des Parameters <i>Wort</i> ersetzt.
<code>\${Parameter:=Wort}</code>	Ist der Parameter <i>Parameter</i> nicht gesetzt oder gleich null, wird er auf den Wert des Parameters <i>Wort</i> gesetzt. Positionsgebundene Parameter können auf diese Weise nicht zugeordnet werden.
<code>\${Parameter:?Wort}</code>	Ist der Parameter <i>Parameter</i> gesetzt und ungleich null, wird sein Wert ersetzt. Andernfalls wird der Wert der Variablen <i>Wort</i> ausgegeben und die Shell verlassen. Wurde die Variable <i>Wort</i> nicht angegeben, wird eine Standardnachricht ausgegeben.
<code>\${Parameter:+Wort}</code>	Ist der Parameter <i>Parameter</i> gesetzt und ungleich null, wird der Wert der Variablen <i>Wort</i> ersetzt.



Eintrag	Beschreibung
<pre> \${Parameter#Muster}   \${Parameter##Muster} </pre>	<p>Stimmt der angegebene Shellparameter <i>Muster</i> mit dem Anfang des Werts für den Parameter <i>Parameter</i> überein, ist der Wert dieser Substitution der Wert des Parameters <i>Parameter</i>, wobei der übereinstimmende Teil gelöscht wird. Andernfalls wird der Wert des Parameters <i>Parameter</i> ersetzt. Mit dem ersten Format wird das kleinste übereinstimmende Muster gelöscht. Mit dem zweiten Format wird das größte übereinstimmende Muster gelöscht.</p>
<pre> \${Parameter%Muster}   \${Parameter%%Muster} </pre>	<p>Wenn das angegebene <i>Shellmuster</i> dem Ende des Wertes der <i>Parametervariablen</i> entspricht, ist der Wert dieser Substitution der Wert der <i>Parametervariablen</i>, aus der der übereinstimmende Teil gelöscht wird. Andernfalls müssen Sie den Wert der <i>Parametervariablen</i> ersetzen. Mit dem ersten Format wird das kleinste übereinstimmende Muster gelöscht. Mit dem zweiten Format wird das größte übereinstimmende Muster gelöscht.</p> <p>In den oben genannten Ausdrücken wird die Variable <i>Wort</i> nur dann ausgewertet, wenn sie als ersetzte Zeichenfolge verwendet werden soll. Demzufolge wird im folgenden Beispiel der Befehl <b>pwd</b> nur ausgeführt, wenn das Flag <b>-d</b> nicht gesetzt oder gleich null ist.</p> <pre>echo \${d:-\$(pwd)}</pre>

**Anmerkung:** Wird der Doppelpunkt (:) in den oben genannten Ausdrücken nicht angegeben, überprüft die Shell nur, ob der Parameter *Parameter* gesetzt ist.

### Zugehörige Konzepte:

„Unbeaufsichtigte Terminals“ auf Seite 312

Alle Systeme bieten Schwachstellen, wenn Terminals mit aktiven Anmeldungen unbeaufsichtigt bleiben. Schwerwiegende Probleme treten auf, wenn der Systemverwalter ein Terminal unbeaufsichtigt lässt, das mit Rootberechtigung aktiviert wurde. Im Allgemeinen sollten sich Benutzer jedes Mal abmelden, wenn sie ihr Terminal verlassen.

*Vordefinierte Sonderparameter in der Korn- bzw. POSIX-Shell:*

Einige Parameter werden automatisch von der Korn- bzw. POSIX-Shell gesetzt.

Die folgenden Parameter werden von der Shell automatisch gesetzt:

Eintrag	Beschreibung
@	<p>Erweitert die positionsgebundenen Parameter, beginnend mit \$1. Jeder Parameter wird durch ein Leerzeichen abgetrennt.</p> <p>Werden doppelte Anführungszeichen (") um @\$ gesetzt, interpretiert die Shell jeden positionsgebundenen Parameter als separate Zeichenfolge. Sind keine positionsgebundenen Parameter vorhanden, erweitert die Shell die Anweisung auf eine nicht in Anführungszeichen gesetzte, leere Zeichenfolge.</p>
*	<p>Erweitert die positionsgebundenen Parameter, beginnend mit \$1. Die Shell trennt jeden Parameter durch das erste Zeichen des Werts für den Parameter <b>IFS</b>.</p> <p>Werden doppelte Anführungszeichen (") um \$* gesetzt, übernimmt die Shell die Werte der positionsgebundenen Parameter in die Anführungszeichen. Die Werte sind durch das erste Zeichen des Parameters <b>IFS</b> getrennt.</p>
#	<p>Gibt die Anzahl (als Dezimalzahl) der an die Shell übergebenen positionsgebundenen Parameter an, wobei der Name der Shellprozedur selbst nicht berücksichtigt wird. Der Parameter \$# liefert demzufolge die Nummer des positionsgebundenen Parameters mit der höchsten Nummer, der gesetzt wurde. Dieser Parameter dient hauptsächlich dazu, zu überprüfen, ob die erforderliche Anzahl an Argumenten vorhanden ist.</p>
-	<p>Liefert der Shell beim Aufruf oder bei Eingabe des Befehls <b>set</b> Flags.</p>
?	<p>Gibt den Exit-Wert des zuletzt ausgeführten Befehls an. Der Wert ist eine Dezimalzeichenfolge. Die meisten Befehle geben den Wert 0 zurück, um die erfolgreiche Beendigung anzuzeigen. Die Shell selbst gibt den aktuellen Wert des Parameters \$? als Exit-Wert zurück.</p>

Eintrag	Beschreibung
\$	Gibt die Prozessnummer der Shell an. Da die Prozessnummern der vorhandenen Prozesse eindeutig sind, wird diese maximal fünfstellige Ziffernfolge oft für die Generierung eindeutiger Namen für temporäre Dateien verwendet.  Das folgende Beispiel verdeutlicht die empfohlene Vorgehensweise bei der Erstellung temporärer Dateien in einem eigens dafür angelegten Verzeichnis: <pre>temp=\$HOME/temp/\$\$ ls &gt;\$temp . . . rm \$temp</pre>
!	Gibt die Prozessnummer des zuletzt aufgerufenen Hintergrundbefehls an.
zero (0)	Erweitert den Namen der Shell oder des Shell-Script.

### Dateinamensubstitution in der Korn- bzw. POSIX-Shell:

Die Korn- bzw. POSIX-Shell führt die Dateinamensubstitution durch, indem sie jedes Befehlswort, das in der Variablen *Wort* angegeben ist, nach bestimmten Zeichen durchsucht.

Wenn ein Befehlswort das Zeichen *\**, *?* oder *[* enthält und das Flag *-f* nicht gesetzt ist, interpretiert die Shell das Wort als Muster. Die Shell ersetzt das Wort durch Dateinamen, die mit dem Muster übereinstimmen und entsprechend der Sortierfolge der verwendeten Sprache sortiert sind. Findet die Shell keinen Dateinamen, der mit dem Muster übereinstimmt, wird das Wort nicht geändert.

Wenn die Shell ein Muster für die Dateinamensubstitution verwendet, muss eine genaue Übereinstimmung mit den Zeichen *.* und */* vorhanden sein.

**Anmerkung:** Die Korn-Shell behandelt diese Zeichen in anderen Fällen der Mustererkennung nicht auf besondere Weise.

Die folgenden Platzhalterzeichen zeigen die entsprechenden Substitutionen an:

Eintrag	Beschreibung
*	Gleicht jede Zeichenfolge ab, einschließlich der leeren Zeichenfolge.
?	Gleicht jedes einzelne Zeichen ab.
[...]	Gleicht jedes der eingeschlossenen Zeichen ab. Mit einem Zeichenpaar, das durch ein Minuszeichen (-) getrennt ist, wird jedes Zeichen, das aufgrund der Sortierfolge in der verwendeten Sprache innerhalb des Bereichs dieses Zeichenpaares liegt, lexikalisch abgeglichen. Ist das erste Zeichen nach der linken eckigen Klammer ([]) ein Ausrufezeichen (!), wird jedes Zeichen, das nicht eingeschlossen ist, abgeglichen. Ein Minuszeichen (-) kann in den Zeichensatz aufgenommen werden, indem es als erstes oder letztes Zeichen angegeben wird.

Die Schreibweise `[:Zeichenklasse:]` kann ebenfalls verwendet werden, um Dateinamen in einem angegebenen Bereich abzugleichen. Mit diesem Format wird das System angewiesen, jedes einzelne Zeichen, das zur Klasse gehört, abzugleichen. Die Definition, welche Zeichen zu einer bestimmten Zeichenklasse gehören, kann über die Kategorie `LC_CTYPE` der Subroutine `setlocale` abgerufen werden. Alle Zeichenklassen, die in der aktuellen Sprache angegeben sind, werden akzeptiert.

Im Folgenden sind die Namen einiger Zeichenklassen aufgeführt:

- **alnum**
- **alpha**
- **cntrl**
- **digit**
- **graph**

- **lower**
- **print**
- **punct**
- **space**
- **upper**
- **xdigit**

Der Schreibweise `[:upper:]` entspricht jeder Großbuchstabe.

Die Korn-Shell unterstützt eine Dateinamenerweiterung, die auf der Sortierung von Elementen oder Symbolen oder auf Äquivalenzklassen basiert.

Eine *Musterliste* ist eine Liste mit Mustern, die durch einen vertikalen Balken (|), das Pipe-Symbol, getrennt sind. Zusammengesetzte Muster können auf folgenden Weise gebildet werden:

Eintrag	Beschreibung
<code>?(PatternList)</code>	Gleicht wahlweise eines der angegebenen Muster ab.
<code>*(Musterliste)</code>	Gleicht kein oder mehr Vorkommen der angegebenen Muster ab.
<code>+(Musterliste)</code>	Gleicht ein oder mehr Vorkommen der angegebenen Muster ab.
<code>@(Musterliste)</code>	Gleicht genau eines der angegebenen Muster ab.
<code>!(PatternList)</code>	Gleicht alle außer einem der angegebenen Muster ab.

Die Mustererkennung unterliegt folgenden Einschränkungen: Ist das erste Zeichen eines Dateinamens ein Punkt (.), kann er nur über ein Muster abgeglichen werden, das auch mit einem Punkt beginnt. Beispielsweise stimmt \* (Stern) mit den Dateinamen `meinedatei` und `ihredatei` überein, aber nicht mit den Dateinamen `.meinedatei` und `.ihredatei`. Um diese Dateien zu finden, müssen Sie ein Muster wie das folgende verwenden:

```
.*datei
```

Stimmt das Muster mit keinem Dateinamen überein, wird das Muster selbst als Ergebnis des versuchten Abgleichs zurückgegeben.

Datei- und Verzeichnisnamen dürfen die Zeichen \*, ?, [ und ] nicht enthalten, da dadurch eine endlose Rekursion (d. h. eine Endlosschleife) während der Abgleichversuche auftreten würde.

*Anführungszeichen entfernen:*

Einige Zeichen werden entfernt, wenn sie nicht in Anführungszeichen gesetzt sind.

Die folgenden Anführungszeichen im ursprünglichen Wort werden entfernt, außer wenn diese selbst in Anführungszeichen eingeschlossen sind: umgekehrter Schrägstrich (\), einfaches Anführungszeichen (') und doppeltes Anführungszeichen (").

### **Eingabe- und Ausgabeumleitung in der Korn- bzw. POSIX-Shell:**

Bevor die Korn-Shell einen Befehl ausführt, sucht sie in der Befehlszeile nach Umleitungszeichen. Mit diesen Sonderzeichen wird die Shell angewiesen, die Eingabe und Ausgabe umzuleiten.

Umleitungszeichen können überall in einem einfachen Befehl auftreten oder vor oder nach einem Befehl stehen. Sie werden nicht an den aufgerufenen Befehl übergeben.

Die Shell führt, außer bei den angegebenen Ausnahmen, eine Befehls- und Parametersubstitution durch, bevor der Parameter **Wort** oder **Ziffer** verwendet wird. Die Dateinamenssubstitution erfolgt nur, wenn das Muster mit einer einzelnen Datei übereinstimmt und keine Interpretation von Leerzeichen durchgeführt wird.

Eintrag	Beschreibung
<Wort	Verwendet die im Parameter <b>Wort</b> angegebene Datei als Standardeingabe (Dateideskriptor 0).
>Wort	Verwendet die im Parameter <b>Wort</b> angegebene Datei als Standardausgabe (Dateideskriptor 1). Ist die Datei nicht vorhanden, wird sie von der Shell erstellt. Ist die Datei vorhanden, und ist die Option <b>noclobber</b> aktiv, tritt ein Fehler auf. Andernfalls wird die Datei auf die Länge null abgeschnitten. <b>Anmerkung:</b> Wenn für mehrere Shells die Option <b>noclobber</b> gesetzt ist und diese Shells ihre Ausgabe an dieselbe Datei umleiten, könnte es zu Konkurrenzsituationen kommen, die unter Umständen dazu führen, dass mehrere Shellprozesse in diese Datei schreiben. Solche Konkurrenzsituationen werden von der Shell nicht erkannt und verhindert.
> Wort	Entspricht dem >Wort mit der Ausnahme, dass mit dieser Umleitungsanweisung die Option <b>noclobber</b> außer Kraft gesetzt wird.
> >Wort	Verwendet die im Parameter <b>Wort</b> angegebene Datei als Standardausgabe (Dateideskriptor 1). Ist die Datei vorhanden, hängt die Shell die Ausgabe an die Datei an (indem sie zuerst an das Dateieinde positioniert). Ist die Datei nicht vorhanden, wird sie von der Shell erstellt.
<>Wort	Öffnet die im Parameter <b>Wort</b> angegebene Datei als Standardeingabe zum Lesen und Schreiben.
<<[-]Wort	Liest jede Zeile der Shellvingabe, bis eine Zeile gefunden wird, die nur den Wert des Parameters <b>Wort</b> oder ein Dateieindezeichen enthält. Die Shell führt keine Parameter-, Befehls- oder Dateinamenssubstitution für die angegebene Datei durch. Das sich daraus ergebende Dokument, das so genannte Here-Dokument, wird zur Standardeingabe. Ist ein Zeichen des Parameters <b>Wort</b> in Anführungszeichen gesetzt, wird keine Interpretation für die Zeichen des Dokuments durchgeführt.

Das *Here*-Dokument wird als Einzelwort behandelt, das nach dem Zeilenvorschubzeichen beginnt und in der Zeile endet, die nur den Begrenzer ohne folgende Leerzeichen enthält. Danach beginnt das nächste Here-Dokument, sofern vorhanden. Das Format ist:

```
[n]<<Wort
Here-DokumentBegrenzer
```

Ist ein Zeichen im *Wort* in Anführungszeichen eingeschlossen, wird der Begrenzer gebildet, indem die Anführungszeichen für das *Wort* entfernt werden. Die Zeilen des *Here*-Dokuments werden nicht erweitert. Andernfalls ist der Begrenzer das *Wort* selbst. Sind keine Zeichen im *Wort* in Anführungszeichen eingeschlossen, werden alle Zeilen des Here-Dokuments für die Parametererweiterung, Befehlssubstitution und arithmetische Erweiterung erweitert.

Die Shell führt die Parametersubstitution für die umgeleiteten Daten durch. Wenn Sie verhindern möchten, dass die Shell die Zeichen \, \$ und ' sowie das erste Zeichen des Parameters **Wort** interpretiert, müssen Sie diesen Zeichen einen Backslash (\) voranstellen.

Ist ein Minuszeichen (-) an <<, angehängt, übergeht die Shell alle führenden Tabulatorzeichen im Parameter **Wort** und im Dokument.

Eintrag	Beschreibung
<&Ziffer	Kopiert die Standardeingabe von dem im Parameter <b>Ziffer</b> angegebenen Dateideskriptor.
>&Ziffer	Kopiert die Standardausgabe in den vom Parameter <b>Ziffer</b> angegebenen Dateideskriptor.
<&-	Schließt die Standardeingabe.
>&-	Schließt die Standardausgabe.
<&p	Überträgt die Eingabe vom Koprozess an die Standardeingabe.
>&p	Überträgt die Ausgabe vom Koprozess an die Standardausgabe.

Ist einer der oben genannten Umleitungsoptionen eine Ziffer vorangestellt, wird die Nummer des Dateideskriptors durch diese Ziffer angegeben (statt der Standardwerte 0 oder 1). Im folgenden Beispiel öffnet die Shell Dateideskriptor 2 zum Schreiben als Kopie von Dateideskriptor 1:

```
... 2>&1
```

Die Reihenfolge, in der Umleitungen angegeben werden, ist entscheidend. Die Shell wertet jede Umleitung nach der Zuordnung (*Dateideskriptor, Datei*) zum Zeitpunkt der Auswertung aus. So wird beispielsweise in der Anweisung

```
... 1>Datei 2>&1
```

der Dateideskriptor 1 der im Parameter **Datei** angegebenen Datei zugeordnet. Die Shell ordnet Dateideskriptor 2 der Datei zu, die dem Dateideskriptor 1 (*Datei*) zugeordnet ist. Bei umgekehrter Reihenfolge der Umleitungen würde Dateideskriptor 2 dem Terminal (angenommen, Dateideskriptor 1 wäre dies vorher gewesen) zugeordnet und Dateideskriptor 1 würde der im Parameter **Datei** angegebenen Datei zugeordnet.

Wenn dem Befehl ein Et-Zeichen (&) nachgestellt und die Jobsteuerung nicht aktiv ist, ist die Standardeingabe für den Befehl die leere Datei /dev/null. Andernfalls enthält die Umgebung für die Ausführung eines Befehls die Dateideskriptoren der aufrufenden Shell, wie von Eingabe- und Ausgabespezifikation geändert.

### Zugehörige Konzepte:

„Eingabe- und Ausgabeumleitung“ auf Seite 362

Im Betriebssystem AIX können Sie die Eingabe und Ausgabe (E/A) von Daten mit bestimmten E/A-Befehlen und -Symbolen steuern.

### Zugehörige Tasks:

„Ausgabe an Inline-Eingabedokumente (Here) umleiten“ auf Seite 366

Sie können Ausgaben an Inline-Eingabedokumente (Here) umleiten.

### Koprozesse:

In der Korn-Shell oder POSIX-Shell ist es möglich, Befehle als Hintergrundprozesse auszuführen. Diese Befehle, die innerhalb eines Shell-Script ausgeführt werden, werden *Koprozesse* genannt.

Wenn Sie hinter einem Befehl den Operator |& setzen, wird dieser Befehl als Koprozess ausgeführt. Die Standardeingabe und die Standardausgabe des Befehls werden über eine Pipe an Ihr Script geleitet.

Ein Koprozess muss die folgenden Rahmenbedingungen erfüllen:

- Am Ende jeder Nachricht muss ein Zeilenvorschubzeichen stehen.
- Jede Ausgabenachricht wird an die Standardausgabe gesendet.
- Nach jeder Nachricht wird der Inhalt der Standardausgabe gelöscht.

Das folgende Beispiel veranschaulicht, wie die Eingabe an einen Koprozess übergeben und von diesem zurückgegeben wird:

```
echo "Initial process"
./FileB.sh |&
read -p a b c d
echo "Read from coprocess: $a $b $c $d"
print -p "Passed to the coprocess"
read -p a b c d
echo "Passed back from coprocess: $a $b $c $d"

FileB.sh
 echo "The coprocess is running"
 read a b c d
 echo $a $b $c $d
```

Das Ergebnis dieses Beispiels ist die folgende Standardausgabe:

```
Initial process
Read from coprocess: The coprocess is running
Passed back from coprocess: Passed to the coprocess
```

Mit dem Befehl **print -p** können Sie in den Koprozess schreiben. Mit dem Befehl **read -p** können Sie aus dem Koprozess lesen.

#### **Zugehörige Konzepte:**

„Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 264

Die Korn-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlsorientierte Programmiersprache. Sie entspricht dem internationalen Standard für Betriebssysteme POSIX (Portable Operating System Interface for Computer Environment).

*Ein- und Ausgabe von Koprozessen umleiten:*

Die Standardeingabe und -ausgabe eines Koprozesses wird über E/A-Umadressierung einem nummerierten Dateideskriptor erneut zugeordnet.

Der Befehl

```
exec 5>&p
```

verschiebt beispielsweise die Eingabe des Koprozesses an den Dateideskriptor 5.

Nach Abschluss dieses Koprozesses können Sie die Standardumleitungssyntax verwenden, um die Befehlsausgabe an den Koprozess umzuleiten. Sie können auch einen weiteren Koprozess starten. Die Ausgabe beider Koprozesse wird mit derselben Pipe verbunden und kann mit dem Befehl **read -p** gelesen werden. Geben Sie zum Stoppen des Koprozesses Folgendes ein:

```
read -u5
```

#### **Integrierte Befehle in der Korn- bzw. POSIX-Shell:**

Sonderbefehle sind in die Korn- bzw. POSIX-Shell integriert und werden im Shellprozess ausgeführt.

Sofern nicht anders angegeben, wird die Ausgabe in den Dateideskriptor 1 geschrieben. Der Exit-Status ist 0 (null), wenn der Befehl keine Syntaxfehler enthält. Die Umleitung von Eingabe und Ausgabe ist erlaubt. Integrierte Befehle sind in zwei Arten unterteilt: *integrierte Sonderbefehle* und *integrierte reguläre Befehle*.

Integrierte Sonderbefehle unterscheiden sich in folgenden Punkten von regulären integrierten Befehlen:

- Ein Syntaxfehler in einem integrierten Sonderbefehl kann dazu führen, dass die Shell, die den Befehl ausführt, beendet wird. Bei einem Syntaxfehler in einem integrierten Befehl ist dies nicht der Fall. Wird das Shellprogramm durch einen Syntaxfehler in einem integrierten Sonderbefehl nicht beendet, ist der Exit-Wert ungleich null.
- Variablenzuordnungen, die mit integrierten Sonderbefehlen angegeben werden, bleiben auch nach Beendigung des Befehls wirksam.
- Umleitungen von Eingabe und Ausgabe werden nach der Parameterzuordnung verarbeitet.

Außerdem werden Wörter, die den Sonderbefehlen **export**, **readonly** und **typeset** in Form einer Parameterzuordnung folgen, nach denselben Regeln erweitert wie eine Parameterzuordnung. Nach dem Gleichheitszeichen (=) wird eine Tildensubstitution vorgenommen. Worttrennung und Dateinamensubstitution werden nicht durchgeführt.

#### **Zugehörige Konzepte:**

„Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 264

Die Korn-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlsorientierte Programmiersprache. Sie entspricht dem internationalen Standard für Betriebssysteme POSIX (Portable Operating System Interface for Computer Environment).

„Funktionen in der Korn-Shell“ auf Seite 268

Das reservierte Wort **function** definiert Shellfunktionen. Die Shell liest und speichert Funktionen intern. Aliasnamen werden beim Lesen der Funktion aufgelöst. Funktionen werden von der Shell auf dieselbe Weise ausgeführt wie Befehle, wobei die Argumente als positionsgebundene Parameter übergeben werden.

#### Zugehörige Verweise:

„Liste der integrierten Sonderbefehle in der Korn- bzw. POSIX-Shell“ auf Seite 230

Sonderbefehle sind in die Korn- bzw. POSIX-Shell integriert und werden im Shellprozess ausgeführt.

„Integrierte reguläre Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 230

Im Folgenden finden Sie eine Liste der integrierten regulären Korn- bzw. POSIX-Shellbefehle.

*Beschreibungen integrierter Sonderbefehle für die Korn- bzw. POSIX-Shell:*

Sonderbefehle sind in die Korn- bzw. POSIX-Shell integriert und werden im Shellprozess ausgeführt.

Die integrierten Sonderbefehle der Korn-Shell sind im Folgenden beschrieben:

:	eval	newgrp	shift	
.		exec	readonly	times
break	exit	return	trap	
continue	export	set	typeset	
			unset	

Eintrag	Beschreibung
: [Argument ...]	Erweitert nur Argumente. Dieser Sonderbefehl wird verwendet, wenn zwar ein Befehl erforderlich ist, dieser Befehl aber keine Aktion ausführen soll, wie beispielsweise in der Bedingung <i>then</i> eines Befehls <b>if</b> .
. Datei [Argument ...]	Liest die angegebene Datei vollständig und führt anschließend die Befehle aus. Die Befehle werden in der aktuellen Shellumgebung ausgeführt. Die Shell verwendet den in der Variablen <i>PATH</i> angegebenen Suchpfad, um das Verzeichnis mit der angegebenen Datei zu finden. Sind Argumente angegeben, werden diese zu positionsgebundenen Parametern. Andernfalls bleiben die positionsgebundenen Parameter unverändert. Der Exit-Status ist der Exit-Status des zuletzt ausgeführten Befehls. Nähere Informationen zu positionsgebundenen Parametern finden Sie im Abschnitt „Parametersubstitution in der Korn- bzw. POSIX-Shell“ auf Seite 240. <b>Anmerkung:</b> Der Befehl <i>.Datei</i> [Argument ...] liest zuerst die gesamte Datei, bevor ein Befehl ausgeführt wird. Daher sind die Befehle <b>alias</b> und <b>unalias</b> in der Datei nicht auf Funktionen anwendbar, die in der Datei definiert sind.
break [n]	Führt mit der Ausführung nach dem nächsten Ende einer <b>for</b> -, <b>while</b> -, <b>until</b> oder <b>select</b> -Schleife fort. Wenn Sie den Parameter <i>n</i> angeben, fährt der Befehl <b>break</b> mit der Ausführung nach dem <i>n</i> -ten Schleifenende fort. Der Wert von <i>n</i> ist eine beliebige ganze Zahl größer-gleich 1.
continue [n]	Nimmt die nächste Iteration durch die Befehlsschleife <b>for</b> , <b>while</b> , <b>until</b> oder <b>select</b> wieder auf. Wenn Sie den Parameter <i>n</i> angeben, setzt der Befehl die Ausführung bei der <i>n</i> -ten Befehlsschleife fort. Der Wert von <i>n</i> ist eine beliebige ganze Zahl größer-gleich 1.
eval [Argument ...]	Liest die angegebenen Argumente als Eingabe für die Shell und führt die entsprechenden Befehle aus.
exec [Argument ...]	Führt anstelle der Shell den im Argument angegebenen Befehl aus, ohne einen neuen Prozess zu erstellen. Sie können Ein- und Ausgabeargumente angeben und damit den aktuellen Prozess beeinflussen. Wenn Sie kein Argument angeben, ändert der Befehl <b>exec</b> die Dateideskriptoren so, wie es in der Liste für die Umleitung von Eingabe und Ausgabe vorgeschrieben ist. In diesem Fall werden alle Dateideskriptoren mit Nummern größer als 2, die auf diese Weise geöffnet wurden, beim Aufruf eines anderen Programms geschlossen.
exit [n]	Beendet die Shell mit dem im Parameter <i>n</i> angegebenen Exit-Status. Der Parameter <i>n</i> muss eine ganze Dezimalzahl ohne Vorzeichen zwischen 0 und 255 sein. Wenn Sie den Parameter <i>n</i> nicht angeben, ist der Exit-Status der Exit-Status des zuletzt ausgeführten Befehls. Sie können die Shell auch durch Angabe eines Dateiendezeichens beenden, sofern die Option <b>ignoreeof</b> des Sonderbefehls <b>set</b> nicht aktiviert ist.

Eintrag	Beschreibung
<b>export -p</b> [ <i>Name</i> [= <i>Wert</i> ]] ...	<p>Markiert die angegebenen Namen für den automatischen Export in die Umgebung von nachfolgend ausgeführten Befehlen.</p> <p>Mit <b>-p</b> werden die Namen und Werte aller exportierten Variablen im folgenden Format in die Standardausgabe geschrieben:  "export %s= %s\n", &lt;Name&gt; &lt;Wert&gt;</p>
<b>newgrp</b> [ <i>Gruppe</i> ]	<p>Äquivalent zum Befehl <b>exec/usr/bin/newgrp</b> [<i>Gruppe</i>].  <b>Anmerkung:</b> Dieser Befehl kehrt nicht zurück.</p>
<b>readonly -p</b> [ <i>Name</i> [= <i>Wert</i> ]] ...	<p>Markiert die im Parameter <i>Name</i> angegebenen Namen als schreibgeschützt. Diese Namen können durch nachfolgende Zuordnung nicht geändert werden.</p> <p>Mit <b>-p</b> werden die Namen und Werte aller exportierten Variablen im folgenden Format in die Standardausgabe geschrieben:  "export %s= %s\n", &lt;Name&gt; &lt;Wert&gt;</p>
<b>return</b> [ <i>n</i> ]	<p>Bewirkt, dass eine Shellfunktion zum aufrufenden Script zurückkehrt. Der Rückkehrstatus wird mit dem Parameter <i>n</i> angegeben. Wenn Sie den Parameter <i>n</i> nicht angeben, ist der Rückkehrstatus der des zuletzt ausgeführten Befehls. Wenn Sie den Befehl <b>return</b> außerhalb einer Funktion oder eines Script aufrufen, hat er dieselbe Wirkung wie der Befehl <b>exit</b>.</p>
<b>set</b> [+   - <b>abCefhkmnostuvx</b> ] [+   - <b>o</b> <i>Option</i> ]... [+   - <b>A</b> <i>Name</i> ] [ <i>Argument</i> ...]	<p>Wenn Sie keine Optionen oder Argumente angeben, schreibt der Befehl <b>set</b> die Namen und Werte aller Shellvariablen in der Sortierfolge der aktuellen Sprache. Bei Angabe von Optionen wird die Zuordnung der Attribute der Shell wie im Folgenden beschrieben festgelegt bzw. aufgehoben.</p> <ul style="list-style-type: none"> <li>-<b>A</b>     Array-Zuordnung. Setzt die Einstellung des Parameters <i>Name</i> außer Kraft und ordnet ihm nacheinander Werte aus der angegebenen Parameterliste <i>Argument</i> zu. Wird das Flag <b>+A</b> verwendet, wird der Parameter <i>Name</i> nicht als erster inaktiviert.</li> <li>-<b>a</b>     Exportiert automatisch alle nachfolgenden definierten Parameter.</li> <li>-<b>b</b>     Asynchrone Meldung über die Beendigung von Hintergrundjobs.</li> <li>-<b>C</b>     Äquivalent zu <b>set -o noclobber</b>.</li> <li>-<b>e</b>     Führt den Trap für ERR aus und beendet die Ausführung bei einem Befehls-Exit-Status ungleich null ist, sofern der einfache Befehl nicht <ul style="list-style-type: none"> <li>+ in einer Liste des Typs <b>&amp;&amp;</b> oder <b>  </b> enthalten ist,</li> <li>+ direkt einer Anweisung <b>if</b>, <b>while</b> oder <b>until</b> folgt,</li> <li>+ in der Pipeline hinter <b>!</b> enthalten ist.</li> </ul> <p>Dieser Modus ist inaktiviert, wenn Profile gelesen werden.</p> </li> <li>-<b>f</b>     Inaktiviert die Dateinamensubstitution.</li> <li>-<b>h</b>     Kennzeichnet jeden Befehl beim ersten Auftreten als Aliasname mit absoluter Pfadangabe.</li> <li>-<b>k</b>     Stellt alle Parameterzuordnungsargumente in die Umgebung eines Befehls, nicht nur die Argumente, die dem Befehlsnamen vorangestellt sind.</li> <li>-<b>m</b>     Führt Hintergrundjobs in einem separaten Prozess aus und gibt beim Abschluss eine Zeile aus. Der Exit-Status von Hintergrundjobs wird in einer Abschlussnachricht mitgeteilt. Bei Systemen mit Jobsteuerung wird dieses Flag automatisch für interaktive Shells aktiviert. Nähere Informationen hierzu finden Sie im Abschnitt „Jobsteuerung in der Korn- bzw. POSIX-Shell“ auf Seite 256.</li> <li>-<b>n</b>     Liest Befehle und überprüft sie auf Syntaxfehler, führt sie aber nicht aus. Dieses Flag wird bei interaktiven Shells ignoriert.</li> </ul>



**-o Option**

Gibt die aktuellen Optionseinstellungen und eine Fehlermeldung aus, wenn kein Argument angegeben wurde. In einer **ksh**-Befehlszeile können mehrere Optionen angegeben werden. Bei Verwendung des Flags **+o** wird die angegebene Option außer Kraft gesetzt. Bei Angabe von Argumenten wird die Zuordnung positionsgebundener Parameter festgelegt bzw. aufgehoben. Die folgenden Argumente können mit der Variablen *Option* angegeben werden:

**allexport**

Entspricht dem Flag **-a**.

**bgnice** Führt alle Hintergrundjobs mit einer niedrigeren Priorität aus. Dies ist der Standardmodus.

**emacs** Öffnet eine Variante des Zeileneditors emacs zur Befehlseingabe.

**errexit** Entspricht dem Flag **-e**.

**gmacs** Öffnet eine Variante des Zeileneditors gmacs zur Befehlseingabe.

**ignoreeof**

Die Shell wird nicht beendet, wenn der Befehl auf ein Dateieindezeichen trifft. Zum Beenden der Shell müssen Sie den Befehl **exit** eingeben oder die Tastenkombination Strg-D mehr als 11 Mal drücken.

**keyword**

Entspricht dem Flag **-k**.

**Anmerkung:** Dieses Flag dient nur der Abwärtskompatibilität mit der Bourne-Shell. Es wird dringend davon abgeraten, dieses Flag in anderen Fällen anzuwenden.

**markdirs**

Hängt einen Backslash (/) an alle Verzeichnisnamen an, die ein Ergebnis der Dateinamenssubstitution sind.

**monitor**

Entspricht dem Flag **-m**.

**noclobber**

Verhindert, dass bei der Umleitung vorhandene Dateien überschrieben werden. Wenn Sie diese Option angeben und die vorhandene Datei überschrieben werden soll, muss dem Umleitungssymbol ein Pipe-Symbol (>|) folgen.

**noexec** Entspricht dem Flag **-n**.

**noglob** Entspricht dem Flag **-f**.

**nolog** Verhindert, dass Funktionsdefinitionen in den Dateien `.profile` und `$ENV` in der Protokolldatei gespeichert werden.

**nounset**

Entspricht dem Flag **-u**.

**privileged**

Entspricht dem Flag **-p**.

Eintrag	Beschreibung
	<p><b>trackall</b> Entspricht dem Flag <b>-h</b>.</p> <p><b>verbose</b> Entspricht dem Flag <b>-v</b>.</p> <p><b>vi</b> Öffnet eine Variante des Zeileneditors vi im Einfügemodus zur Befehlseingabe. Mit der Eingabe des Escape-Zeichens 033 wird der Übertragungsmodus für den Editor aufgerufen. Durch Drücken der Eingabetaste wird die Zeile übergeben.</p> <p><b>viraw</b> Verarbeitet jedes Zeichen so, wie im vi-Modus eingegeben.</p> <p><b>xtrace</b> Entspricht dem Flag <b>-x</b>.</p> <p><b>-p</b> Verhindert die Verarbeitung der Datei <code>\$HOME/.profile</code> und verwendet die Datei <code>/etc/suid_profile</code> anstelle der Datei <code>ENV</code>. Dieser Modus wird jedes Mal inaktiviert, wenn die aktuelle Benutzer-ID (UID) oder Gruppen-ID (GID) nicht mit der realen UID oder GID übereinstimmt. Durch Inaktivieren dieser Option wird die aktuelle UID oder GID auf die reale UID bzw. GID gesetzt.</p> <p><b>Anmerkung:</b> Die Option <b>-p</b> wird vom System nicht unterstützt, da das Betriebssystem die Shell-Scripts <b>setuid</b> nicht unterstützt.</p> <p><b>-s</b> Sortiert die positionsgebundenen Parameter lexikografisch.</p> <p><b>-t</b> Bricht die Ausführung nach dem Lesen und der Ausführung eines Befehls ab.</p> <p><b>Anmerkung:</b> Dieses Flag dient nur der Abwärtskompatibilität mit der Bourne-Shell. Es wird dringend davon abgeraten, dieses Flag in anderen Fällen anzuwenden.</p> <p><b>-u</b> Behandelt nicht gesetzte Parameter bei der Substitution als Fehler.</p> <p><b>-v</b> Gibt Shellvingabezeilen aus, wenn sie gelesen werden.</p> <p><b>-x</b> Gibt Befehle und die zugehörigen Argumente aus, wenn sie ausgeführt werden.</p> <p><b>-</b> Inaktiviert die Flags <b>-x</b> und <b>-v</b> und stoppt die Überprüfung der Argumente auf Flags.</p> <p><b>—</b> Verhindert, dass Flags geändert werden. Die Verwendung dieser Option empfiehlt sich, wenn der Parameter <code>\$1</code> auf einen Wert gesetzt ist, der mit einem Minuszeichen (<b>-</b>) beginnt. Folgen auf dieses Flag keine Argumente, werden die positionsgebundenen Parameter nicht gesetzt.</p> <p>Wird einem der Flags des Befehls <b>set</b> ein Pluszeichen (+) anstelle eines Minuszeichens (-) vorangestellt, wird das Flag inaktiviert. Diese Flags können beim Aufruf der Shell verwendet werden. Wenn der Befehl 'set +o' ohne Argumente aufgerufen wird, zeigt er die aktuellen Einstellungen an. Die Ausgabe erfolgt so, dass sie sich als Befehlsfolge für die Wiederherstellung der aktuellen Einstellung eignet. Die aktuelle Gruppe von Flags befindet sich im Parameter <b>\$-</b>. Falls Sie das Flag <b>-A</b> nicht angeben, sind die verbleibenden Argumente positionsgebundene Parameter und werden in der angegebenen Reihenfolge <code>\$1</code>, <code>\$2</code>, ... usw. zugeordnet. Werden keine Argumente angegeben, werden die Namen und Werte aller benannten Parameter in der Standardausgabe ausgegeben.</p>
<b>shift</b> [ <i>n</i> ]	Benennt die positionsgebundenen Parameter <code>\$n+1 ...</code> bis <code>\$1 ...</code> um. Der Standardwert des Parameters <i>n</i> ist 1. Der Parameter <i>n</i> kann jeder arithmetische Ausdruck sein, dessen Ergebnis eine nicht negative Zahl kleiner-gleich <code>\$#</code> ist.
<b>times</b>	Gibt die aufgelaufenen Benutzer- und Systemzeiten für die Shell und die von der Shell ausgeführten Prozesse an.

Eintrag	Beschreibung
<b>trap</b> [ <i>Befehl</i> ] [ <i>Signal</i> ] ...	<p>Führt den angegebenen Befehl aus, wenn die Shell das bzw. die angegebenen Signale empfängt. Der Parameter <i>Befehl</i> wird einmal gelesen, wenn der Trap gesetzt wird, und ein weiteres Mal, wenn der Trap ausgeführt wird. Für den Parameter <i>Signal</i> kann die Nummer oder der Name des Signals angegeben werden. Trap-Befehle werden in der Reihenfolge der Signalnummern ausgeführt. Jeder Versuch, einen Trap für ein Signal zu setzen, das beim Starten der aktuellen Shell ignoriert wurde, ist wirkungslos.</p> <p>Wenn Sie den Befehl - angeben, werden alle Traps auf die ursprünglichen Werte zurückgesetzt. Falls Sie keinen Befehl angeben und das erste Signal numerisch angegeben wurde, setzt der Befehl <b>ksh</b> den Wert für die Parameter <i>Signal</i> auf die ursprünglichen Werte zurück.</p> <p><b>Anmerkung:</b> Wenn kein Befehl und für das erste Signal ein symbolischer Name angegeben ist, wird das Signal als Befehl interpretiert.</p> <p>Hat der Parameter <i>Signal</i> den Wert <b>ERR</b>, wird der angegebene Befehl jedes Mal ausgeführt, wenn ein Befehl einen Exit-Status ungleich null hat. Wenn der Parameter den Wert <b>DEBUG</b> hat, wird der angegebene Befehl nach jedem Befehl ausgeführt. Hat der Parameter <i>Signal</i> den Wert <b>0</b> oder <b>EXIT</b> und wird der Befehl <b>trap</b> innerhalb einer Funktion ausgeführt, wird der angegebene Befehl nach Abschluss der Funktion ausgeführt. Wenn der Parameter <i>Signal</i> den Wert <b>0</b> oder <b>EXIT</b> für einen Befehl <b>trap</b> hat, der außerhalb einer Funktion definiert ist, wird der angegebene Befehl bei Beendigung der Shell ausgeführt.</p> <p><b>Anmerkung:</b> Wenn ein Script ein SIGINT -Signal in einer Funktion empfängt, kann das EXIT-Signal nicht abgefangen werden, wenn die Shell beendet wird.</p> <p>Der Befehl <b>trap</b> ohne Argumente gibt eine Liste mit Befehlen aus, die jeder Signalnummer zugeordnet sind. Wenn der angegebene Befehl NULL ist (wird angezeigt als "" (leere Anführungszeichen)), ignoriert der Befehl <b>ksh</b> das Signal. Weitere Informationen dazu, wie die Korn-Shell bzw. die POSIX-Shell ein Zeichen als reguläres Zeichen liest, finden Sie im Abschnitt „Zeichen in der Korn- bzw. POSIX-Shell in Anführungszeichen setzen“ auf Seite 232.</p> <p>Eine vollständige Liste der gültigen Werte für den Parameter <i>Signal</i>, die mit dem Befehl <b>trap</b> ohne das Präfix <b>SIG</b> angegeben werden können, finden Sie in den Beschreibungen der Subroutinen <b>sigaction</b>, <b>sigvec</b> und <b>signal</b> in der Veröffentlichung <i>Technical Reference: Base Operating System and Extensions, Volume 2</i>.</p>

Eintrag	Beschreibung
<b>typeset</b> [+HLRZfiltrux [ <i>n</i> ]] [ <i>Name</i> [= <i>Wert</i> ]] ...	<p>Setzt Attribute und Werte für Shellparameter. Wird der Befehl innerhalb einer Funktion aufgerufen, wird ein neues Exemplar des Parameters <i>Name</i> erstellt. Parameterwert und -typ werden zurückgeschrieben, wenn die Funktion beendet ist. Die folgenden Flags können mit dem Befehl <b>typeset</b> angegeben werden:</p> <ul style="list-style-type: none"> <li>-H Ermöglicht die Zuordnung von AIX- zu Hostdateien auf Maschinen, auf denen AIX nicht ausgeführt wird.</li> <li>-L Richtet den Parameter <i>Wert</i> linksbündig aus und entfernt führende Leerzeichen. Hat der Parameter <i>n</i> einen Wert ungleich null, definiert dieser die Feldlänge. Andernfalls wird die Feldlänge durch die Länge des Werts der ersten Zuordnung bestimmt. Bei der Zuordnung des Parameters wird dieser rechts mit Leerzeichen gefüllt oder, falls erforderlich, abgeschnitten, damit er in das Feld passt. Führende Nullen werden entfernt, wenn zusätzlich das Flag -Z gesetzt wurde. Das Flag -R wird inaktiviert.</li> <li>-R Richtet den Parameter rechtsbündig aus und füllt das Feld mit führenden Leerzeichen auf. Hat der Parameter <i>n</i> einen Wert ungleich null, definiert dieser die Feldlänge. Andernfalls wird die Feldlänge durch die Länge des Werts der ersten Zuordnung bestimmt. Das Feld bleibt mit Leerzeichen aufgefüllt oder wird vom Ende her abgeschnitten, wenn der Parameter erneut zugeordnet wird. Das Flag L wird inaktiviert.</li> <li>-Z Richtet den Parameter rechtsbündig aus und füllt das Feld mit führenden Nullen auf, wenn es sich bei dem ersten Zeichen, das kein Leerzeichen ist, um eine Ziffer handelt und das Flag -L nicht gesetzt wurde. Hat der Parameter <i>n</i> einen Wert ungleich null, definiert dieser die Feldlänge. Andernfalls wird die Feldlänge durch die Länge des Werts der ersten Zuordnung bestimmt.</li> <li>-f Gibt an, dass die Namen auf Funktionen und nicht auf Parameter verweisen. Es können keine Zuordnungen vorgenommen werden, und die einzigen weiteren gültigen Flags sind -t, -u und -x. Mit dem Flag -t wird der Ausführungstrace für diese Funktion aktiviert. Das Flag -u bewirkt, dass diese Funktion als nicht definiert markiert wird. Die Variable <i>FPATH</i> wird durchsucht, um die Funktionsdefinition zu finden, wenn auf die Funktion verwiesen wird. Das Flag -x bewirkt, dass Funktionsdefinitionen in Shell-Scripts gültig bleiben, die nicht mit einem separaten Aufruf des Befehls <b>ksh</b> gestartet werden.</li> <li>-i Gibt den Parameter als ganze Zahl an und beschleunigt damit die Berechnung. Hat der Parameter <i>n</i> einen Wert ungleich null, definiert er die Berechnungsbasis für die Ausgabe. Andernfalls wird die Ausgabebasis durch die erste Zuordnung festgelegt.</li> <li>-l Wandelt alle Großbuchstaben in Kleinbuchstaben um. Das Flag -u für die Umwandlung in Großbuchstaben wird inaktiviert.</li> <li>-r Markiert die im Parameter <i>Name</i> angegebenen Namen als schreibgeschützt. Diese Namen können durch nachfolgende Zuordnung nicht geändert werden.</li> <li>-t Bezeichnet die benannten Parameter. Bezeichnungen können vom Benutzer definiert werden und haben keine Sonderbedeutung für die Shell.</li> <li>-u Wandelt alle Kleinbuchstaben in Großbuchstaben um. Das Flag -l für die Umwandlung in Kleinbuchstaben wird inaktiviert.</li> <li>-x Markiert den im Parameter <i>Name</i> angegebenen Namen für den automatischen Export in die Umgebung von nachfolgend ausgeführten Befehlen.</li> </ul> <p>Die Verwendung eines Pluszeichens + anstelle eines Minuszeichens - inaktiviert die Flags des Befehls <b>typeset</b>. Sind keine Parameter <i>Name</i>, aber Flags angegeben, wird eine Liste mit Namen (und wahlweise mit Werten) der Parameter ausgegeben, bei denen diese Flags gesetzt sind. (Die Verwendung eines Pluszeichens (+) anstelle eines Minuszeichens (-) verhindert, dass die Werte ausgegeben werden.) Werden keine Namen und keine Flags angegeben, werden die Namen und Attribute aller Parameter ausgegeben.</p>
<b>unset</b> [-fv ] <i>Name</i> ...	<p>Inaktiviert die Werte und Attribute der Parameter, die in der Namensliste angegeben sind. Bei Angabe von -v bezieht sich <i>Name</i> auf den Variablenamen. Der Name wird von der Shell inaktiviert und aus der Umgebung entfernt. Schreibgeschützte Variablen können nicht inaktiviert werden. Durch Inaktivieren der Variablen <i>ERRNO</i>, <i>LINENO</i>, <i>MAILCHECK</i>, <i>OPTARG</i>, <i>OPTIND</i>, <i>RANDOM</i>, <i>SECONDS</i>, <i>TMOUT</i> und Unterstrich (<u>  </u>) geht die Sonderbedeutung dieser Variablen verloren, auch wenn sie nachfolgend zugeordnet werden.</p> <p>Ist das Flag -f gesetzt, bezieht sich <i>Name</i> auf einen Funktionsnamen. Die Funktionsdefinition wird von der Shell inaktiviert.</p>

## Beschreibungen integrierter regulärer Befehle für die Korn- bzw. POSIX-Shell:

Im Folgenden sind die integrierten Befehle für die Korn- bzw. POSIX-Shell beschrieben.

Die Korn-Shell stellt die folgenden integrierten regulären Befehle bereit:

alias	fg	print	ulimit	
bg	getopts	pwd	umask	
cd	jobs	read	unalias	
command	kill	setgroups	wait	
echo	let	setenv	test	whence
fc				

Eintrag	Beschreibung
<b>alias</b> [-t ] [-x ] [Aliasname [= Zeichenfolge]]	Mit diesem Befehl werden Aliasnamen erstellt oder erneut definiert oder vorhandene Aliasnamen in die Standardausgabe geschrieben.
...	Weitere Informationen finden Sie in der Beschreibung des Befehls <b>alias</b> in der Veröffentlichung "Commands Reference".
<b>bg</b> [Job-ID...]	Setzt den angegebenen Job im Hintergrund fort. Der aktuelle Job wird im Hintergrund fortgesetzt, wenn kein Parameter <i>Job-ID</i> angegeben ist. Nähere Informationen zur Jobsteuerung finden Sie im Abschnitt „Jobsteuerung in der Korn- bzw. POSIX-Shell“ auf Seite 256.  Weitere Informationen zur Ausführung des Befehls <b>jobs</b> im Hintergrund finden Sie in der Beschreibung des Befehls <b>bg</b> in der Veröffentlichung "Commands Reference".
<b>cd</b> [Argument] <b>cd</b> <i>Alt Neu</i>	Für diesen Befehl werden zwei Formate unterstützt. Mit dem ersten Format wird vom aktuellen Verzeichnis in das im Parameter <i>Argument</i> angegebene Verzeichnis gewechselt. Ist der mit dem Parameter <i>Argument</i> angegebene Wert ein Minuszeichen (-), wird in das vorherige Verzeichnis gewechselt. Die Shellvariable <b>HOME</b> ist der Standardwert für den Parameter <i>Argument</i> . Die Variable <b>PWD</b> wird auf das aktuelle Verzeichnis gesetzt.  Die Shellvariable <b>CDPATH</b> definiert den Suchpfad für das Verzeichnis, das den Wert des Parameters <i>Argument</i> enthält. Alternative Verzeichnisnamen werden durch einen Doppelpunkt (:) voneinander getrennt. Der Standardpfad ist leer. Damit wird das aktuelle Verzeichnis angegeben. Das aktuelle Verzeichnis wird durch einen leeren Pfadnamen angegeben, der unmittelbar nach dem Gleichheitszeichen oder zwischen begrenzenden Doppelpunkten an einer beliebigen Stelle in der Pfadliste steht. Beginnt das angegebene Argument mit einem Schrägstrich (/), wird der Suchpfad nicht verwendet. Andernfalls wird jedes Verzeichnis im Pfad nach dem Argument durchsucht.  Die zweite Form des Befehls <b>cd</b> ersetzt die Zeichenfolge der Variablen <i>Neu</i> durch die Zeichenfolge der Variablen <i>Alt</i> im aktuellen Verzeichnis <b>PWD</b> und versucht, in dieses neue Verzeichnis zu wechseln.
<b>command</b> [-p ] <i>Befehlsname</i> [Argument ...] <b>command</b> [-v   -V ] <i>Befehlsname</i>	Mit diesem Befehl wird die Shell angewiesen, den angegebenen Befehl und die Argumente als einfachen Befehl auszuführen. Die Suche nach Shellfunktionen wird damit unterdrückt.  Weitere Informationen finden Sie in der Beschreibung des Befehls <b>command</b> in der Veröffentlichung "Commands Reference".
<b>echo</b> [Zeichenfolge ...]	Schreibt Zeichenfolgen in die Standardausgabe. Nähere Informationen zur Verwendung finden Sie in der Beschreibung des Befehls <b>echo</b> . Das Flag <b>-n</b> wird nicht unterstützt.
<b>fc</b> [-r ] [-e <i>Editor</i> ] [ <i>Erster</i> [ <i>Letzter</i> ]] <b>fc</b> -l [-n ] [-r ] [ <i>Erster</i> [ <i>Letzter</i> ]]	
<b>fc</b> -s [ <i>Alt= Neu</i> ] [ <i>Erster</i> ]	Zeigt den Inhalt Ihrer Befehlsprotokolldatei an oder ruft einen Editor auf, in dem Sie die zuvor in der Shell eingegebenen Befehle ändern und erneut ausführen können.  Weitere Informationen finden Sie in der Beschreibung des Befehls <b>fc</b> in der Veröffentlichung "Commands Reference".

Eintrag	Beschreibung
<b>fg</b> [ <i>Job-ID</i> ]	<p>Setzt die Ausführung des angegebenen Jobs im Vordergrund fort. Wenn Sie keine Jobs angeben, setzt der Befehl den aktuellen Job im Vordergrund fort.</p> <p>Weitere Informationen zur Ausführung von Jobs im Vordergrund finden Sie in der Beschreibung des Befehls <b>fg</b> in der Veröffentlichung "Commands Reference".</p>
<b>getopts</b> Optionszeichenfolge <i>Name</i> [ <i>Argument ...</i> ]	<p>Überprüft den Parameter <i>Argument</i> auf gültige Optionen.</p> <p>Weitere Informationen finden Sie in der Beschreibung des Befehls <b>getopts</b> in der Veröffentlichung "Commands Reference".</p>
<b>jobs</b> [-l   -n   -p] [ <i>Job-ID ...</i> ]	<p>Zeigt den Status von Jobs an, die in der aktuellen Shellumgebung gestartet wurden. Wird für den Parameter <i>Job-ID</i> kein bestimmter Job angegeben, werden die Statusinformationen für alle aktiven Jobs angezeigt. Wenn die Beendigung eines Jobs gemeldet wird, entfernt die Shell die Prozess-ID dieses Jobs aus der Jobliste für die aktuelle Shellumgebung.</p> <p>Weitere Informationen finden Sie in der Beschreibung des Befehls <b>jobs</b> in der Veröffentlichung "Commands Reference".</p>
<b>kill</b> [-s { <i>Signalname</i>   <i>Signalnummer</i> } ] <i>Prozess-ID...</i> <b>kill</b> [- <i>Signalname</i>   - <i>Signalnummer</i> ] <i>Prozess-ID...</i> <b>kill</b> -l [ <i>Exit-Status</i> ]	<p>Sendet ein Signal (standardmäßig das Signal <b>SIGTERM</b>) an einen aktiven Prozess. Diese Standardaktion stoppt Prozesse normalerweise. Wenn Sie einen Prozess stoppen möchten, müssen Sie die Prozess-ID (PID) in der Variablen <i>Prozess-ID</i> angeben. Die Shell meldet die PID jedes Prozesses, der im Hintergrund ausgeführt wird, sofern Sie nicht mehrere Prozesse in einer Befehlskette starten. In diesem Fall meldet die Shell nur die Nummer des letzten Prozesses. Sie können mit dem Befehl <b>ps</b> auch die Prozess-ID von Befehlen feststellen.</p> <p>Listet Signalnamen auf.</p> <p>Weitere Informationen finden Sie in der Beschreibung des Befehls <b>kill</b> in der Veröffentlichung "Commands Reference".</p>
<b>let</b> <i>Ausdruck ...</i>	<p>Wertet die angegebenen arithmetischen Ausdrücke aus. Der Exit-Status ist 0, wenn der Wert des letzten Ausdrucks ungleich null ist. Weitere Informationen finden Sie im Abschnitt „Arithmetische Berechnungen in der Korn- bzw. POSIX-Shell“ auf Seite 228.</p>
<b>print</b> [-Rnprsu [ <i>n</i> ]] [ <i>Argument ...</i> ]	<p>Druckt die Shellausgabe. Wenn Sie keine Flags angeben oder ein Minuszeichen (-) oder ein doppeltes Minuszeichen (--) angeben, werden die Argumente, wie beim Befehl <b>echo</b> beschrieben, in der Standardausgabe ausgegeben. Die Flags haben folgende Funktionen:</p> <ul style="list-style-type: none"> <li><b>-R</b> Gibt eine unformatierte Ausgabe aus (die Escape-Konventionen für den Befehl <b>echo</b> werden ignoriert). Das Flag <b>-R</b> gibt alle nachfolgenden Argumente und Flags außer <b>-n</b> aus.</li> <li><b>-n</b> Verhindert, dass der Ausgabe ein Zeilenvorschubzeichen hinzugefügt wird.</li> <li><b>-p</b> Schreibt die Argumente nicht in die Standardausgabe, sondern in die Pipe des Prozesses, der mit  &amp; (Pipe-Symbol mit Et-Zeichen) ausgeführt wird.</li> <li><b>-r</b> Gibt eine unformatierte Ausgabe aus. Die Escape-Konventionen für den Befehl <b>echo</b> werden ignoriert.</li> <li><b>-s</b> Schreibt die Argumente nicht in die Standardausgabe, sondern in das Befehlsprotokoll.</li> <li><b>-u</b> Gibt eine einstellige Einheitennummer <i>n</i> für den Dateideskriptor an, in den die Ausgabe gestellt wird. Der Standardwert ist 1.</li> </ul>
<b>pwd</b>	<p>Entspricht dem Befehl <code>print -r - \$PWD</code>.</p> <p><b>Anmerkung:</b> Der interne Korn-Shellbefehl <b>pwd</b> unterstützt keine symbolischen Verbindungen.</p>
<b>read</b> [-prsu [ <i>n</i> ]] [ <i>Name?Aufforderung</i> ] [ <i>Name...</i> ]	<p>Liest die Shellvingabe. Jeweils eine Zeile wird gelesen und in Felder aufgeteilt, wobei die Zeichen der Variablen <b>IFS</b> als Trennzeichen verwendet werden.</p> <p>Weitere Informationen finden Sie in der Beschreibung des Befehls <b>read</b> in der Veröffentlichung "Commands Reference".</p>
<b>setgroups</b>	<p>Führt den Befehl <code>/usr/bin/setgroups</code> aus, der in einer separaten Shell ausgeführt wird. Nähere Informationen zu diesem Befehl finden Sie in der Beschreibung des Befehls <b>setgroups</b>. Es besteht jedoch ein Unterschied. Mit dem integrierten Befehl <b>setgroups</b> wird eine Subshell aufgerufen. Mit dem Befehl <b>setgroups</b> wird die aktuell ausgeführte Shell ersetzt. Da der integrierte Befehl nur aus Gründen der Kompatibilität unterstützt wird, wird empfohlen, in Scripts den absoluten Pfadnamen <code>/usr/bin/setgroups</code> anstelle des integrierten Shellbefehls zu verwenden.</p>

Eintrag	Beschreibung
<b>setsenv</b>	Führt den Befehl <code>/usr/bin/setsenv</code> aus, der die derzeit ausgeführte Shell ersetzt. Nähere Informationen zu diesem Befehl finden Sie in der Beschreibung des Befehls <b>setsenv</b> .
<b>test</b>	Entspricht <i>[Ausdruck]</i> . Syntaxinformationen und eine Beschreibung dieses Befehls finden Sie im Abschnitt „Bedingungsausdrücke in der Korn- bzw. POSIX-Shell“ auf Seite 231.
<b>ulimit [-HSacdfmst ]</b> <i>[Grenzwert]</i>	<p>Definiert und zeigt die Ressourcengrenzen für Benutzerprozesse an, die in der Datei <code>/etc/security/limits</code> definiert sind. Die Datei enthält die folgenden Standardgrenzwerte:</p> <pre> fsize = 2097151 core = 2048 cpu = 3600 data = 131072 rss = 65536 stack = 8192 threads = -1 </pre> <p>Diese Werte werden als Standardeinstellung verwendet, wenn dem System ein Benutzer hinzugefügt wird. Die Werte werden mit dem Befehl <b>mkuser</b> festgelegt, wenn der Benutzer dem System hinzugefügt wird, oder mit dem Befehl <b>chuser</b> geändert.</p> <p>Grenzwerte können in die Kategorien veränderlich oder fest eingeteilt werden. Benutzer können veränderliche Grenzwerte mit dem Befehl <b>ulimit</b> bis zum maximalen festen Grenzwert ändern. Feste Grenzwerte können nur mit der Rootberechtigung geändert werden.</p> <p>Viele Systeme arbeiten entweder mit nur einem oder mit gar keinem dieser Grenzwerte. Der Grenzwert für eine bestimmte Ressource wird mit dem Parameter <i>Grenzwert</i> definiert. Der Wert für den Parameter <i>Grenzwert</i> kann eine Zahl in der Maßeinheit, die für jede Ressource angegeben ist, oder der Wert <code>unlimited</code> (uneingeschränkt) sein. Folgende Flags können mit dem Befehl <b>ulimit</b> angegeben werden:</p> <ul style="list-style-type: none"> <li><b>-H</b> Gibt an, dass der feste Grenzwert für eine bestimmte Ressource gesetzt ist. Mit Rootberechtigung kann der feste Grenzwert erhöht werden. Der Grenzwert kann von jedem Benutzer heruntersetzt werden.</li> <li><b>-S</b> Gibt an, dass der veränderliche Grenzwert für eine bestimmte Ressource gesetzt ist. Ein veränderlicher Grenzwert kann bis auf den Wert des festen Grenzwerts erhöht werden. Wenn weder die Option <b>-H</b> noch die Option <b>-S</b> angegeben ist, gilt der Grenzwert für beide Kategorien.</li> <li><b>-a</b> Listet alle aktuellen Ressourcengrenzen auf.</li> <li><b>-c</b> Gibt die Größe des Kernspeicherauszugs mit der Anzahl von 512-Byte-Blöcken an.</li> <li><b>-d</b> Gibt die Größe des Datenbereichs in Kilobyte an.</li> <li><b>-f</b> Gibt die Anzahl der 512-Byte-Blöcke für Dateien an, die von untergeordneten Prozessen geschrieben wurden (Dateien beliebiger Größe können gelesen werden).</li> <li><b>-m</b> Gibt die Größe des physischen Speichers in Kilobyte an.</li> <li><b>-n</b> Gibt den Grenzwert für die Anzahl der Dateideskriptoren an, die durch einen Prozess geöffnet werden können.</li> <li><b>-r</b> Gibt den Grenzwert für die Anzahl der Threads pro Prozess an.</li> <li><b>-s</b> Gibt die Größe des Stapelspeicherbereichs in Kilobyte an.</li> <li><b>-t</b> Gibt die Anzahl der Sekunden an, die von jedem Prozess verwendet werden dürfen.</li> </ul> <p>Wird die Variable <i>Grenzwert</i> nicht angegeben, wird die aktuelle Ressourcengrenze ausgegeben. Der veränderliche Grenzwert wird ausgegeben, wenn das Flag <b>-H</b> nicht angegeben wird. Werden mehrere Ressourcen angegeben, werden der Grenzwertname und die Maßeinheit vor dem Wert ausgegeben. Wird keine Option angegeben, wird das Flag <b>-f</b> angenommen. Wird dieser Wert geändert, empfiehlt es sich, für die Variable <i>Grenzwert</i> einen unveränderlichen und einen veränderlichen Grenzwert zu definieren, sofern die Flags <b>-H</b> oder <b>-S</b> nicht angegeben werden.</p> <p>Weitere Informationen zu den Grenzwerten für Benutzer und Systemressourcen finden Sie in den Beschreibungen der Subroutinen <b>getrlimit</b>, <b>setrlimit</b> und <b>vlimit</b>.</p>

Eintrag	Beschreibung
<b>umask</b> [-S ] [ <i>Maske</i> ]	<p>Legt die Dateiberechtigungen fest. Beim Erstellen einer Datei werden mit diesem Wert zusammen mit den Berechtigungen des Erstellungsprozesses die Berechtigungen der Datei festgelegt. Der Standardwert ist 022. Wird der Parameter <i>Maske</i> nicht angegeben, zeigt der Befehl <b>umask</b> die Erstellungsmaske für den Dateimodus der aktuellen Shellumgebung in der Standardausgabe an.</p> <p>Weitere Informationen zu Dateiberechtigungen finden Sie in der Beschreibung des Befehls <b>umask</b> in der Veröffentlichung "Commands Reference".</p>
<b>unalias</b> { -a   <i>Aliasname...</i> }	<p>Entfernt die angegebene Aliasdefinition oder entfernt alle Aliasdefinitionen, wenn das Flag <b>-a</b> angegeben ist. Die Aliasdefinitionen werden aus der aktuellen Shellumgebung entfernt.</p> <p>Weitere Informationen finden Sie in der Beschreibung des Befehls <b>unalias</b> in der Veröffentlichung "Commands Reference".</p>
<b>wait</b> [ <i>Prozess-ID...</i> ]	<p>Wartet auf den angegebenen Job und wird dann beendet. Wenn Sie keinen Job angeben, wartet der Befehl auf alle momentan aktiven untergeordneten Prozesse. Der Exit-Status dieses Befehls entspricht dem des Prozesses, auf den der Befehl wartet.</p> <p>Weitere Informationen finden Sie in der Beschreibung des Befehls <b>wait</b> in der Veröffentlichung "Commands Reference".</p>
<b>whence</b> [-pv ] <i>Name ...</i>	<p>Zeigt für jeden angegebenen Namen an, wie die Shell ihn als Befehlsnamen interpretieren würde. Bei der Verwendung ohne ein Flag zeigt <b>whence</b> den absoluten, allen Namen entsprechenden Pfadnamen an, falls ein solcher vorhanden ist.</p> <p><b>-p</b> Führt eine Pfadsuche für die angegebenen Namen durch, selbst wenn es sich dabei um Aliasnamen, Funktionen oder reservierte Wörter handelt.</p> <p><b>-v</b> Erzeugt einen ausführlichen Bericht, in dem der Typ jedes Namens angegeben wird.</p>

## Jobsteuerung in der Korn- bzw. POSIX-Shell:

Die Korn- bzw. POSIX-Shell stellt eine Möglichkeit zur Steuerung von Befehlsfolgen oder *Jobs* zur Verfügung.

Wird der Sonderbefehl **set -m** ausgeführt, ordnet die Korn-Shell jeder Befehlskette einen Job zu. Die Shell führt eine Tabelle der aktuellen Jobs, die mit dem Befehl **jobs** ausgegeben wird, und ordnet ihnen kleine, ganze Zahlen mit normalem Wertebereich zu.

Wenn Sie einen Job mit einem Et-Zeichen (&) als Hintergrundjob starten, gibt die Shell eine Zeile wie die folgende aus:

```
[1] 1234
```

Damit wird angezeigt, dass der im Hintergrund gestartete Job die Nummer 1 hat. Außerdem wird angezeigt, dass der Job einen (Ausgangs-)Prozess mit der Prozess-ID 1234 hatte.

Ein Job kann mit der Tastenkombination Strg-Z abgebrochen werden. Mit dieser Tastenkombination wird ein Signal **STOP** an den aktuellen Job gesendet. Die Shell zeigt normalerweise an, dass der Job gestoppt wurde, und zeigt anschließend die Shellvingabeaufforderung an. Sie können den Jobstatus ändern (die weitere Ausführung des Jobs mit dem Befehl **fg** im Hintergrund fortsetzen), andere Befehle ausführen und die Ausführung des Jobs anschließend mit dem Befehl **fg** wieder im Vordergrund fortsetzen. Die Tastenkombination Strg-Z hat eine unmittelbare Wirkung und ist insofern eine Unterbrechung, als die Shell anstehende Ausgaben und ungelesene Eingaben löscht, wenn die Tastenkombination gedrückt wird.

Ein im Hintergrund ausgeführter Job wird gestoppt, wenn er versucht, vom Terminal zu lesen. Die Generierung von Ausgaben durch Hintergrundjobs ist normalerweise zulässig. Sie können diese Option mit dem Befehl **stty tostop** inaktivieren. Wenn Sie diese Terminaloption setzen, werden Hintergrundjobs gestoppt, wenn sie versuchen, eine Ausgabe zu erzeugen oder eine Eingabe zu lesen.



In der Korn-Shell können Sie auf verschiedene Arten auf Jobs verweisen. Für den Verweis auf einen Job kann die Prozess-ID jedes seiner Prozesse oder eine der folgenden Methoden verwendet werden:

Eintrag	Beschreibung
%Nummer	Gibt den Job mit der ihm zugeordneten Nummer an.
%Zeichenfolge	Gibt jeden Job an, dessen Befehlszeile mit der Variablen <i>Zeichenfolge</i> beginnt.
%%?Zeichenfolge	Gibt jeden Job an, dessen Befehlszeile die Variable <i>Zeichenfolge</i> enthält.
%%	Gibt den aktuellen Job an.
%+	Äquivalent zu %%.
%-	Gibt den vorherigen Job an.

Diese Shell erkennt Änderungen des Prozessstatus sofort. Normalerweise gibt sie eine Nachricht aus, wenn der Job so blockiert ist, dass kein Fortschritt mehr möglich ist. Die Nachricht erfolgt kurz vor der Ausgabe einer Eingabeaufforderung, so dass die Arbeit des Benutzers nicht gestört wird.

Ist der Überwachungsmodus aktiv, löst jeder beendete Hintergrundjob die Traps aus, die für das Signal **CHLD** gesetzt wurden.

Wenn Sie versuchen, die Shell zu verlassen (durch Eingabe von `exit` oder Drücken der Tastenkombination Strg-D), während Jobs gestoppt sind oder ausgeführt werden, gibt das System die Warnung Es wurden (aktive) Jobs gestoppt aus. Mit dem Befehl `jobs` kann überprüft werden, welche Jobs betroffen sind. Wenn Sie den Beendigungsversuch sofort wiederholen, beendet die Shell die gestoppten und aktiven Jobs ohne Warnung.

*Signalverarbeitung:*

Die Signale **SIGINT** und **SIGQUIT** für einen aufgerufenen Befehl werden ignoriert, wenn dem Befehl ein Et-Zeichen (&) nachgestellt ist und die Jobüberwachungsoption **monitor** nicht aktiv ist. Andernfalls haben die Signale die Werte, die die Shell vom übergeordneten Prozess übernommen hat.

Wenn ein Signal, für das ein Trap definiert ist, empfangen wird, während die Shell auf die Beendigung eines Vordergrundbefehls wartet, wird der dem Signal zugeordnete Trap erst nach Beendigung des Vordergrundbefehls ausgeführt. Daher wird ein Trap für ein Signal **CHILD** erst ausgeführt, wenn der Vordergrundjob beendet ist.

### Befehlszeileneditierung in der Korn- bzw. POSIX-Shell:

Normalerweise geben Sie eine Befehlszeile an einem Terminal ein und beenden sie mit einem Zeilenvorschubzeichen (**Eingabetaste** oder **Zeilenvorschub**). Wenn Sie die Befehlszeileneditierungsoption `emacs`, `gmacs` oder `vi` aktivieren, können Sie die Befehlszeile editieren.

Mit den folgenden Befehlen werden im Editiermodus aufgerufen:

Eintrag	Beschreibung
<code>set -o emacs</code>	Wechselt in den <code>emacs</code> -Editiermodus und startet eine Variante des Zeileneditors <code>emacs</code> .
<code>set -o gmacs</code>	Wechselt in den <code>gmacs</code> -Editiermodus und startet eine Variante des Zeileneditors <code>gmacs</code> .
<code>set -o vi</code>	Wechselt in den <code>vi</code> -Editiermodus und startet eine Variante des Zeileneditors <code>vi</code> .

Eine Editieroption wird jedes Mal automatisch ausgewählt, wenn der Variablen `VISUAL` oder `EDITOR` ein Wert zugeordnet wird, der mit einem dieser Optionsnamen endet.

**Anmerkung:** Um die Editierfunktionen zu verwenden, muss das Terminal die Eingabetaste als Wagenrücklauf ohne Zeilenvorschub akzeptieren. Das aktuelle Zeichen in der Anzeige muss mit einem Leerzeichen überschrieben werden.

In jedem Editiermodus wird ein Fenster an der aktuellen Zeile geöffnet. Die Fensterbreite entspricht dem Wert der Variablen *COLUMNS*, falls diese definiert ist. Andernfalls beträgt die Fensterbreite 80 Zeichen. Ist die Zeile länger als die Fensterbreite minus zwei, macht das System den Benutzer durch eine Markierung am Ende des Fensters darauf aufmerksam. Während der Cursor bewegt wird und den Fensterrand erreicht, wird das Fenster um den Cursor zentriert. Folgende Markierungen werden angezeigt:

Eintrag	Beschreibung
>	Gibt an, dass die Zeile über den rechten Rand des Fensters hinausgeht.
<	Gibt an, dass die Zeile über den linken Rand des Fensters hinausgeht.
*	Gibt an, dass die Zeile über beide Fensterränder hinausgeht.

Mit den im jeweiligen Editiermodus verfügbaren Suchbefehlen können Sie auf die Protokolldatei der Korn-Shell zugreifen. Es werden nur Zeichenfolgen verglichen. Wenn das führende Zeichen in der Zeichenfolge ein Winkelzeichen (^) ist, muss die Übereinstimmung mit dem ersten Zeichen in der Zeile beginnen.

### Zugehörige Konzepte:

„Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 264

Die Korn-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlsorientierte Programmiersprache. Sie entspricht dem internationalen Standard für Betriebssysteme POSIX (Portable Operating System Interface for Computer Environment).

*Editiermodus emacs:*

Der Editiermodus emacs wird aufgerufen, wenn Sie die Option **emacs** oder **gmacs** aktivieren. Der einzige Unterschied zwischen diesen beiden Modi besteht in der Verarbeitung des Editierbefehls Strg-T.

Wenn Sie eine Befehlszeile editieren möchten, positionieren Sie den Cursor an der Stelle, die Sie korrigieren möchten, und fügen je nach Bedarf Zeichen bzw. Wörter ein oder löschen sie. Alle Editierbefehle sind Steuerzeichen oder Escape-Zeichenfolgen.

Editierbefehle funktionieren an jeder Position einer Zeile und nicht nur am Anfang der Zeile. Nach Editierbefehlen dürfen, abgesehen von den angegebenen Ausnahmen, die Eingabetaste oder die Taste für den Zeilenvorschub nicht gedrückt werden.

Eintrag	Beschreibung
Strg-F	Bewegt den Cursor um ein Zeichen weiter vor (nach rechts).
Esc-F	Bewegt den Cursor ein Wort (eine Zeichenfolge, die nur aus Buchstaben, Ziffern und Unterstreichungszeichen besteht) weiter.
Strg-B	Bewegt den Cursor ein Zeichen zurück (nach links).
Esc-B	Bewegt den Cursor ein Wort zurück.
Strg-A	Bewegt den Cursor an den Anfang der Zeile.
Strg-E	Bewegt den Cursor an das Ende der Zeile.
Strg-] c	Bewegt den Cursor in der aktuellen Zeile vorwärts bis zum angegebenen Zeichen.
Esc-Strg-] c	Bewegt den Cursor in der aktuellen Zeile rückwärts bis zum angegebenen Zeichen.
Strg-X Strg-X	Tauscht den Cursor gegen die Markierung aus.
ERASE	Löscht das vorherige Zeichen. (Benutzerdefiniertes Löschanzeigezeichen, das mit dem Befehl <b>stty</b> definiert wird, normalerweise die Tastenkombination Strg-H.)
Strg-D	Löscht das aktuelle Zeichen.
Esc-D	Löscht das aktuelle Wort.
Esc-Rücktaste	Löscht das vorherige Wort.

Eintrag	Beschreibung
Esc-H	Löscht das vorherige Wort.
Esc-Löschtaste	Löscht das vorherige Wort. Ist die Löschtaste als Unterbrechungszeichen definiert, ist dieser Befehl nicht gültig.
Strg-T	Setzt das aktuelle Zeichen zusammen mit dem nächsten Zeichen im Modus emacs um. Setzt die zwei vorherigen Zeichen im Modus gmacs um.
Strg-C	Setzt das aktuelle Zeichen in Großbuchstaben.
Esc-C	Setzt das aktuelle Wort in Großbuchstaben.
Esc-L	Setzt das aktuelle Wort in Kleinbuchstaben.
Strg-K	Löscht die Zeichen vom Cursor bis zum Ende der Zeile. Wird ein numerischer Parameter vorangestellt, dessen Wert kleiner ist als die aktuelle Cursorposition, werden mit diesem Editierbefehl die Zeichen von der angegebenen Position bis zum Cursor gelöscht. Wird ein numerischer Parameter vorangestellt, dessen Wert größer ist als die aktuelle Cursorposition, werden mit diesem Editierbefehl die Zeichen vom Cursor bis zur angegebenen Position gelöscht.
Strg-W	Löscht die Zeichen vom Cursor bis zur Markierung.
Esc-P	Stellt den Bereich vom Cursor bis zur Markierung in den Puffer.
KILL	Benutzerdefiniertes Löschrzeichen, das mit dem Befehl <b>stty</b> definiert wird, normalerweise die Tastenkombination Strg-G oder ein At-Zeichen (@). Löscht die gesamte aktuelle Zeile. Werden zwei Löschrzeichen nacheinander eingegeben, erfolgt nach allen nachfolgenden Löschrzeichen ein Zeilenvorschub (empfiehlt sich bei Fernschreibern).
Strg-Y	Stellt den letzten aus der Zeile gelöschten Eintrag wieder her. (Stellt den Eintrag aus dem Speicher zurück in die Zeile.)
Strg-L	Bewirkt einen Zeilenvorschub und die Ausgabe der aktuellen Zeile.
Strg-@	(Nullzeichen) Setzt eine Markierung.
Esc-Leerschrift	Setzt eine Markierung.
Strg-J	(Neue Zeile) Führt die aktuelle Zeile aus.
Strg-M	(Eingabetaste) Führt die aktuelle Zeile aus.
EOF	Verarbeitet das Dateiendezeichen (normalerweise die Tastenkombination Strg-D) nur als Dateiende, wenn die aktuelle Zeile leer ist.
Strg-P	Ruft den vorherigen Befehl ab. Bei jedem Drücken der Tastenkombination Strg-P wird auf den zuletzt eingegebenen Befehl zugegriffen. Bewegt den Cursor eine Zeile zurück, wenn er sich nicht in der ersten Zeile eines Befehls mit mehreren Zeilen befindet.
Esc-<	Ruft die zeitlich am weitesten zurückliegende (älteste) Protokollzeile ab.
Esc->	Ruft die zeitlich am nächsten zurückliegende (jüngste) Protokollzeile ab.
Strg-N	Ruft die nächste Befehlszeile ab. Bei jedem Drücken der Tastenkombination Strg-N wird auf den zeitlich nachfolgenden Befehl zugegriffen.
Strg-R <i>Zeichenfolge</i>	Keht die Richtung für die Suche einer früheren Befehlszeile im Protokoll um, die die mit dem Parameter <b>Zeichenfolge</b> angegebene Zeichenfolge enthält. Ist der Wert null (0) angegeben, wird vorwärts gesucht. Die angegebene Zeichenfolge wird durch Drücken der Eingabetaste oder ein Zeilenvorschubzeichen abgeschlossen. Falls der Zeichenfolge ein Winkelzeichen (^) vorangestellt ist, muss eine übereinstimmende Zeile mit dem Parameter <b>Zeichenfolge</b> beginnen. Ist der Parameter <b>Zeichenfolge</b> nicht angegeben, wird auf die nächste Befehlszeile zugegriffen, die den jüngsten Parameter <b>Zeichenfolge</b> enthält. In diesem Fall kehrt der Wert 0 die Suchrichtung um.
Strg-O	(Operate) Führt die aktuelle Zeile aus und ruft die der aktuellen Zeile nachfolgende nächste Zeile aus der Protokolldatei auf.
Esc <i>Ziffern</i>	(Escape) Definiert den numerischen Parameter. Die Ziffern werden als Parameter für den nächsten Befehl übernommen. Die Befehle, die einen Parameter akzeptieren, sind <b>Strg-F</b> , <b>Strg-B</b> , <b>ERASE</b> , <b>Strg-C</b> , <b>Strg-D</b> , <b>Strg-K</b> , <b>Strg-R</b> , <b>Strg-P</b> , <b>Strg-N</b> , <b>Strg-I</b> , <b>Esc-</b> , <b>Esc-Strg-I</b> , <b>Esc-</b> , <b>Esc-B</b> , <b>Esc-C</b> , <b>Esc-D</b> , <b>Esc-F</b> , <b>Esc-H</b> , <b>Esc-L</b> und <b>Esc-Strg-H</b> .

Eintrag	Beschreibung
<b>Esc</b> <i>Buchstabe</i>	(Programmfunktionssymbol) Sucht in der Aliasliste nach einem Aliasnamen <i>_Buchstabe</i> . Ist ein solcher Aliasname definiert, wird dessen Wert in die Eingabewarteschlange gestellt. Der Parameter <i>Buchstabe</i> darf keine der Escape-Funktionen angeben.
<b>Esc-</b> [ <i>Buchstabe</i> ]	(Programmfunktionssymbol) Sucht in der Aliasliste nach einem Aliasnamen <i>__Buchstabe</i> (doppelter Unterstreichungsstrich). Ist ein solcher Aliasname definiert, wird dessen Wert in die Eingabewarteschlange gestellt. Dieser Befehl kann auf vielen Terminals dazu verwendet werden, Funktionsstasten zu definieren.
<b>Esc-</b> .	Fügt in der Zeile das letzte Wort des vorherigen Befehls ein. Ist dem Befehl ein numerischer Parameter vorangestellt, legt der Wert dieses Parameters fest, welches Wort anstelle des letzten Worts eingefügt werden soll.
<b>Esc-</b> _	Entspricht der Tastenkombination <b>Esc-</b> .
<b>Esc-</b> *	Startet die Dateinamensubstitution für das aktuelle Wort. Ein Stern (*) wird angehängt, wenn das Wort mit keiner Datei übereinstimmt oder besondere Musterzeichen enthält.
<b>Esc-Esc</b>	Dateinamenvervollständigung. Ersetzt das aktuelle Wort durch das längste gemeinsame Präfix aller Dateinamen, die mit dem aktuellen Wort übereinstimmen, und hängt einen Stern an. Ist die Übereinstimmung eindeutig, wird ein Schrägstrich (/) angehängt, wenn es sich bei der Datei um ein Verzeichnis handelt. Es wird ein Leerzeichen angehängt, wenn es sich bei der Datei nicht um ein Verzeichnis handelt.
<b>Esc=</b>	Listet die Dateien, die mit dem aktuellen Wortmuster übereinstimmen, so auf, als wäre ein Stern (*) angehängt.
<b>Strg-U</b>	Multipliziert den Parameter des nächsten Befehls mit 4.
<b>\</b>	Setzt die Sonderbedeutung des nächsten Zeichens außer Kraft. Editierzeichen und die Zeichen für <b>ERASE</b> , <b>KILL</b> und <b>INTERRUPT</b> (normalerweise die Löschtaste) können in einer Befehlszeile oder in einem Suchbegriff verwendet werden, wenn ihnen ein Backslash (\) vorangestellt wird. Durch den Backslash werden die möglicherweise vorhandenen Editierfunktionen des Folgezeichens ignoriert.
<b>Strg-V</b>	Zeigt die Shellversion an.
<b>Esc-</b> #	Fügt ein Nummernzeichen (#) am Anfang der Zeile ein und führt anschließend die Zeile aus. Dadurch wird ein Kommentar in der Protokolldatei eingefügt.

### Editiermodus *vi*:

Der Editiermodus *vi* verfügt über zwei Eingabemodi.

Die Modi sind:

- **Eingabemodus.** Wenn Sie einen Befehl eingeben, arbeitet der Editor *vi* im Eingabemodus.
- **Steuermodus.** Drücken Sie die Taste **Esc**, um in den Steuermodus zu wechseln.

Die meisten Steuerbefehle akzeptieren einen optionalen Wiederholungsparameter **Anzahl** vor dem Befehl. Auf den meisten Systemen wird im Modus *vi* zunächst die kanonische Verarbeitung aktiviert. Der Befehl wird zurückgemeldet, wenn mindestens eine der folgenden Bedingungen erfüllt ist:

- Die Geschwindigkeit liegt bei 1200 Baud oder höher.
- Der Befehl enthält Steuerzeichen.
- Es ist weniger als eine Sekunde seit der Ausgabe der Eingabeaufforderung vergangen.

Das Abbruchzeichen (**Esc**) beendet die kanonische Verarbeitung für den Rest des Befehls. Danach kann die Befehlszeile geändert werden. Dieses Schema bietet die Vorteile der kanonischen Verarbeitung mit einem Eingabepufferecho für unformatierten Modus. Wenn auch die Option **viraw** gesetzt ist, ist die kanonische Verarbeitung grundsätzlich aktiviert. Dieser Modus ist bei Systemen implizit, die nicht zwei alternative Begrenzer für Zeilenende unterstützen, und kann für bestimmte Terminals nützlich sein.

Die verfügbaren Editierbefehle von vi sind in Kategorien eingeteilt. Die Kategorien sind im Folgenden aufgeführt:

*Befehle für die Eingabeeditierung:*

Die Befehle für die Eingabeeditierung in der Korn-Shell sind im Folgenden beschrieben.

**Anmerkung:** Standardmäßig arbeitet der Editor im Eingabemodus.

Eintrag	Beschreibung
<b>ERASE</b>	Löscht das vorherige Zeichen. (Benutzerdefiniertes Löschanzeigezeichen, das mit dem Befehl <b>stty</b> definiert wird, normalerweise die Tastenkombination Strg-H oder #.)
<b>Strg-W</b>	Löscht das vorherige, durch Leerzeichen abgetrennte Wort.
<b>Strg-D</b>	Beendet die Shell.
<b>Strg-V</b>	Setzt die Sonderbedeutung des nächsten Zeichens außer Kraft. Editierzeichen, wie z. B. <b>ERASE</b> und <b>KILL</b> , können in der Befehlszeile oder in einem Suchbegriff eingegeben werden, wenn die Tastenkombination Strg-V vorangestellt wird. Mit der Tastenkombination Strg-V werden die Editierfunktionen des nächsten Zeichens, falls vorhanden, gelöscht.
<b>\</b>	Setzt die Sonderbedeutung des nächsten Zeichens <b>ERASE</b> und <b>KILL</b> außer Kraft.

*Editierbefehle für Cursorbewegung:*

Im Folgenden sind die Editierbefehle für Cursorbewegung in der Korn-Shell beschrieben.

Mit folgenden Editierbefehlen kann der Cursor bewegt werden:

Eintrag	Beschreibung
<b>[Anzahl]l</b>	Bewegt den Cursor um ein Zeichen weiter vor (nach rechts).
<b>[Anzahl]w</b>	Bewegt den Cursor ein alphanumerisches Wort weiter.
<b>[Anzahl]W</b>	Bewegt den Cursor an den Anfang des nächsten Wortes, das auf ein Leerzeichen folgt.
<b>[Anzahl]e</b>	Bewegt den Cursor an das Ende des aktuellen Wortes.
<b>[Anzahl]E</b>	Bewegt den Cursor an das Ende des aktuellen, durch Leerzeichen abgetrennten Wortes.
<b>[Anzahl]h</b>	Bewegt den Cursor ein Zeichen zurück (nach links).
<b>[Anzahl]b</b>	Bewegt den Cursor ein Wort zurück.
<b>[Anzahl]B</b>	Bewegt den Cursor zum vorherigen, durch Leerzeichen abgetrennten Wort.
<b>[Anzahl]l</b>	Bewegt den Cursor in die im Parameter <i>Anzahl</i> angegebene Spalte.
<b>[Anzahl]fc</b>	Sucht das nächste Zeichen <i>c</i> in der aktuellen Zeile.
<b>[Anzahl]Fc</b>	Sucht das vorherige Zeichen <i>c</i> in der aktuellen Zeile.
<b>[Anzahl]tc</b>	Äquivalent zu <b>f</b> , gefolgt von <b>h</b> .
<b>[Anzahl]Tc</b>	Äquivalent zu <b>F</b> , gefolgt von <b>l</b> .
<b>[Anzahl];</b>	Wiederholt den letzten Suchbefehl für Einzelzeichen ( <b>f</b> , <b>F</b> , <b>t</b> oder <b>T</b> ) so oft, wie im Parameter <i>Anzahl</i> angegeben ist.
<b>[Anzahl],</b>	Keht den letzten Suchbefehl für Einzelzeichen so oft um, wie im Parameter <i>Anzahl</i> angegeben ist.
<b>0</b>	Bewegt den Cursor an den Anfang einer Zeile.
<b>^</b>	Bewegt den Cursor auf das erste Zeichen einer Zeile, das weder ein Leerzeichen noch ein Tabulatorzeichen ist.
<b>\$</b>	Bewegt den Cursor an das Ende einer Zeile.

## Sucheditierbefehle:

Sucheditierbefehle greifen wie folgt auf Ihr Befehlsprotokoll zu:

Eintrag	Beschreibung
[Anzahl]k	Ruft den vorherigen Befehl ab.
[Anzahl]-	Äquivalent zum Befehl k.
[Anzahl]j	Ruft den nächsten Befehl ab. Bei jeder Eingabe des Befehls j wird auf den nächsten Befehl zugegriffen.
[Anzahl]+	Äquivalent zum Befehl j.
[Anzahl]G	Ruft den Befehl ab, dessen Nummer durch den Parameter <i>Anzahl</i> angegeben ist. Standardmäßig ist der älteste Befehl history eingestellt.
/Zeichenfolge	Sucht im Protokoll rückwärts nach einem früheren Befehl, der die angegebene Zeichenfolge enthält. Die Zeichenfolge wird durch Drücken der <b>Eingabetaste</b> oder ein Zeilenvorschubzeichen beendet. Ist der angegebenen Zeichenfolge ein Winkelzeichen (^) vorangestellt, muss die übereinstimmende Zeile mit dem Parameter <i>Zeichenfolge</i> beginnen. Ist der Wert des Parameters <i>Zeichenfolge</i> gleich null, wird die vorherige Zeichenfolge verwendet.
?Zeichenfolge	Entspricht / <i>Zeichenfolge</i> , außer dass die Suche vorwärts ausgeführt wird.
n	Sucht nach der nächsten Übereinstimmung mit dem letzten Muster der Befehle / <i>Zeichenfolge</i> oder ?.
N	Sucht nach der nächsten Übereinstimmung mit dem letzten Muster der Befehle / <i>Zeichenfolge</i> oder ?, allerdings in entgegengesetzter Richtung. Durchsucht das Protokoll nach der Zeichenfolge, die im vorherigen Befehl / <i>Zeichenfolge</i> eingegeben wurde.

## Editierbefehle für Textänderung:

Editierbefehle für die Textänderung ändern die Zeile:

Eintrag	Beschreibung
a	Aktiviert den Eingabemodus. Die Texteingabe beginnt hinter dem aktuellen Zeichen.
A	Hängt den Text am Ende der Zeile an. Äquivalent zum Befehl \$a.
[Anzahl]c <i>Bewegung</i> c[Anzahl] <i>Bewegung</i>	Löscht alle Zeichen vom aktuellen Zeichen bis zu dem Zeichen, das durch mit dem Parameter <i>Bewegung</i> als Cursorposition angegeben ist, und aktiviert den Eingabemodus. Wenn der Parameter <i>Bewegung</i> den Wert c hat, wird die gesamte Zeile gelöscht und der Eingabemodus aktiviert.
C	Löscht alle Zeichen vom aktuellen Zeichen bis zum Ende der Zeile und aktiviert den Eingabemodus. Äquivalent zum Befehl c\$.
S	Äquivalent zum Befehl cc.
D	Löscht alle Zeichen vom aktuellen Zeichen bis zum Ende der Zeile. Äquivalent zum Befehl d\$.
Eintrag	Beschreibung
[Anzahl]d <i>Bewegung</i> d[Anzahl] <i>Bewegung</i>	Löscht alle Zeichen vom aktuellen Zeichen bis einschließlich des im Parameter <i>Bewegung</i> angegebenen Zeichens. Hat der Parameter <i>Bewegung</i> den Wert d, wird die gesamte Zeile gelöscht.
i	Aktiviert den Eingabemodus. Die Texteingabe beginnt vor dem aktuellen Zeichen.
I	Fügt Text vor dem ersten Zeichen der Zeile ein. Äquivalent zum Befehl 0i.
[Anzahl]P	Wiederholt die letzte Textänderung vor der aktuellen Cursorposition.
[Anzahl]p	Wiederholt die letzte Textänderung hinter der aktuellen Cursorposition.
R	Aktiviert den Eingabemodus und überschreibt die Zeichen in der Anzeige.

Eintrag	Beschreibung
[Anzahl]rc	Ersetzt die mit dem Parameter <i>Anzahl</i> angegebene Anzahl von Zeichen ab der aktuellen Cursorposition durch die mit dem Parameter <i>c</i> angegebenen Zeichen. Nach dem Ersetzen der Zeichen wird der Cursor weiter positioniert.
[Anzahl]x	Löscht das aktuelle Zeichen.
[Anzahl]X	Löscht das vorhergehende Zeichen.
[Anzahl].	Wiederholt der vorherigen Befehl für Textänderung.
[Anzahl]~	Keht die Groß-/Kleinschreibung bei der im Parameter <i>Anzahl</i> angegebenen Anzahl von Zeichen ab der aktuellen Cursorposition um, und setzt den Cursor nach vorn.
[Anzahl]_	Hängt das mit dem Parameter <i>Anzahl</i> des vorherigen Befehls angegebene Wort an und aktiviert den Eingabemodus. Ist der Parameter <i>Anzahl</i> nicht angegeben, wird das letzte Wort verwendet.
*	Hängt einen Stern (*) an das aktuelle Wort an und versucht eine Dateinamenssubstitution. Wird keine Übereinstimmung gefunden, ertönt ein Signalton. Andernfalls wird das Wort durch das passende Muster ersetzt und der Eingabemodus aktiviert.
\	Dateinamenvervollständigung. Ersetzt das aktuelle Wort durch das längste gemeinsame Präfix aller Dateinamen, die mit dem aktuellen Wort übereinstimmen, und hängt einen Stern (*) an. Wenn es nur eine Übereinstimmung gibt, wird ein Schrägstrich (/) angehängt, wenn die Datei ein Verzeichnis ist. Es wird ein Leerzeichen angehängt, wenn die Datei kein Verzeichnis ist.

*Verschiedene Editierbefehle:*

Im Folgenden sind gängige Editierbefehle beschrieben.

Eintrag	Beschreibung
[Anzahl]y <i>Bewegung</i>	
y[Anzahl] <i>Bewegung</i>	Stellt Zeichen, beginnend beim aktuellen Zeichen bis hin zu der mit dem Parameter <i>Bewegung</i> angegebenen Cursorposition (einschließlich) in den Löschpuffer. Text und Cursor bleiben unverändert.
Y	Stellt alle Zeichen von der aktuellen Position bis zum Ende der Zeile in den Speicher. Entspricht dem Befehl y\$.
u	Hebt den letzten Befehl für Textänderung auf.
U	Hebt alle für die entsprechende Zeile ausgeführten Befehle für Textänderung auf.
[Anzahl]v	Gibt den Befehl fc -e \${VISUAL:-\${EDITOR:-vi}} <i>Anzahl</i> in den Eingabepuffer zurück. Wird der Parameter <i>Anzahl</i> nicht angegeben, wird die aktuelle Zeile verwendet.
Strg-L	Bewirkt einen Zeilenvorschub und die Ausgabe der aktuellen Zeile. Dieser Befehl gilt nur im Steuermodus.
Strg-J	(Neue Zeile) Führt die aktuelle Zeile unabhängig vom Modus aus.
Strg-M	(Eingabetaste) Führt die aktuelle Zeile unabhängig vom Modus aus.
#	Sendet die Zeile nach dem Einfügen eines Nummernzeichens (#) am Anfang der Zeile. Die Verwendung dieses Befehls empfiehlt sich, wenn Sie die aktuelle Zeile in das Protokoll einfügen möchten, ohne sie auszuführen.  Wenn die Befehlszeile ein Pipe-Symbol ( ) oder ein Semikolon oder ein Zeilenvorschubzeichen enthält, werden vor jedem dieser Symbole zusätzliche Nummernzeichen (#) eingefügt. Zum Löschen aller Nummernzeichen rufen Sie die Befehlszeile aus dem Protokoll ab und geben ein weiteres Nummernzeichen (#) ein.
=	Listet die Dateinamen, die mit dem aktuellen Muster übereinstimmen, so auf, als wäre ein Stern angehängt.
@ <i>Buchstabe</i>	Sucht in der Aliasliste nach einem Aliasnamen <i>_Buchstabe</i> . Ist ein solcher Aliasname definiert, wird dessen Wert zur Verarbeitung in die Eingabewarteschlange gestellt.

## Befehle in der Korn- bzw. POSIX-Shell:

Die Korn-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlsorientierte Programmiersprache. Sie entspricht dem internationalen Standard für Betriebssysteme POSIX (Portable Operating System Interface for Computer Environment).

Bei POSIX handelt es sich nicht um ein Betriebssystem, sondern um einen *Standard* für die Portierbarkeit von Anwendungen auf Quellenebene auf eine Vielzahl von Systemen. Die POSIX-Funktionen stützen sich auf die Korn-Shell. Die Korn-Shell (oder POSIX-Shell) enthält viele Funktionen, die auch von der Bourne-Shell und der C-Shell zur Verfügung gestellt werden, wie z. B. E/A-Umadressierung, Variablen- und Dateinamensubstitution. Außerdem enthält sie verschiedene zusätzliche Befehle und Funktionen der Programmiersprachen:

**Anmerkung:** Es ist eine eingeschränkte Version der Korn-Shell verfügbar, die Restricted Shell oder **rksh**. Nähere Einzelheiten hierzu finden Sie in der Beschreibung des Befehls **rksh**.

Eintrag	Beschreibung
<b>Arithmetische Berechnung</b>	<p>Die Korn- bzw. POSIX-Shell kann mit dem integrierten Befehl <b>let</b> Rechenoperationen mit ganzen Zahlen und einer Basis von 2 bis 36 ausführen.</p> <p>Führen Sie die folgenden Befehle aus, damit in der Korn-Shell Zahlen erkannt werden, die mit 0 (oktal) und 0x (hexadezimal) beginnen:</p> <p><b>export XPG_SUS_ENV=ON</b> Das Exportieren der Variablen XPG_SUS_ENV bewirkt, dass die ausgeführten Befehle und die von den Befehlen verwendeten Bibliotheken vollständig mit POSIX kompatibel sind. <b>Anmerkung:</b> Da das gesamte Bibliothekssystem mit POSIX kompatibel wird, kann sich das erwartete Standardverhalten bestimmter Befehle ändern.</p> <p><b>export OCTAL_CONST=ON</b> Das Exportieren dieser Variablen bewirkt, dass die Interpretation von Konstanten, die in der Korn-Shell deklariert werden, insofern mit POSIX kompatibel ist, als es die Erkennung oktaler und hexadezimaler Konstanten betrifft.</p>
<b>Befehlsprotokoll</b>	<p>Die Korn- bzw. POSIX-Shell speichert eine Datei, in der alle eingegebenen Befehle aufgezeichnet werden. Mit einem Texteditor kann ein Befehl aus dieser Protokolldatei geändert und anschließend erneut ausgeführt werden.</p>
<b>Koprozesse</b>	<p>Koprozesse bieten Ihnen die Möglichkeit, Programme im Hintergrund auszuführen und Informationen an diese Hintergrundprozesse zu senden und von diesen zu empfangen.</p>
<b>Editieren</b>	<p>Die Korn- bzw. POSIX-Shell stellt Optionen für Befehlszeileneditierung zur Verfügung, mit denen die Befehlszeile editiert werden kann. Dazu stehen Texteditoren wie emacs, gmacs und vi zur Verfügung.</p>

Die folgenden Befehlsarten sind in der Korn-Shell verfügbar:

- Einfacher Befehl
- Befehlskette
- Liste
- Zusammengesetzter Befehl
- Funktion

Wenn Sie in der Korn- bzw. POSIX-Shell einen Befehl ausführen, wertet die Shell den Befehl aus und führt folgende Operationen durch:

- Sie nimmt alle angegebenen Substitutionen vor.
- Sie stellt fest, ob der Befehl einen Schrägstrich (/) enthält. Ist das der Fall, führt die Shell das Programm aus, das im angegebenen Pfadnamen enthalten ist.

Enthält der Befehl keinen Schrägstrich (/), führt die Korn- bzw. POSIX-Shell folgende Aktionen aus:

- Sie stellt fest, ob der Befehl ein spezieller integrierter Befehl ist. Ist das der Fall, führt die Shell den Befehl innerhalb des aktuellen Shellprozesses aus.



- Sie vergleicht den Befehl mit den benutzerdefinierten Funktionen. Stimmt der Befehl mit einer benutzerdefinierten Funktion überein, werden die positionsgebundenen Parameter gespeichert und anschließend auf die Argumente des *Funktionsaufrufs* gesetzt. Wenn die Funktion beendet ist oder eine Nachricht zurückgibt, wird die Liste mit den positionsgebundenen Parametern zurückgeschrieben, und alle Traps, die innerhalb der Funktion auf EXIT gesetzt sind, werden ausgeführt. Der Wert einer Funktion ist der Wert des zuletzt ausgeführten Befehls. Eine Funktion wird im aktuellen Shellprozess ausgeführt.
- Stimmt der Befehlsname mit dem Namen eines integrierten Befehls überein, wird der entsprechende integrierte Befehl aufgerufen.
- Sie erstellt einen Prozess und versucht, den Befehl mit **exec** auszuführen (sofern der Befehl kein integrierter und kein benutzerdefinierter Befehl ist).

Die Korn- bzw. POSIX-Shell durchsucht jedes Verzeichnis in einem angegebenen Pfad nach ausführbaren Dateien. Die Shellvariable *PATH* definiert den Suchpfad für das Verzeichnis, das den Befehl enthält. Alternative Verzeichnisnamen werden durch einen Doppelpunkt (:) voneinander getrennt. Der Standardpfad ist */usr/bin:*, wobei zuerst das Verzeichnis */usr/bin* und dann das aktuelle Verzeichnis angegeben wird. Das aktuelle Verzeichnis wird durch mindestens zwei benachbarte Doppelpunkte oder durch einen Doppelpunkt am Anfang und Ende der Pfadliste angegeben.

Falls die Datei Ausführungsberechtigung hat, aber kein Verzeichnis oder eine Datei vom Typ *a.out* ist, nimmt die Shell an, dass sie Shellbefehle enthält. Der aktuelle Shellprozess erzeugt zum Lesen der Datei eine Subshell. Alle nicht exportierten Aliasnamen, Funktionen und benannten Parameter werden aus der Datei gelöscht. Falls die Shellbefehlsdatei Leseberechtigung (*read*) hat oder das Bit **setuid** oder **setgid** für die Datei gesetzt sind, führt die Shell einen Agenten aus, der die Berechtigungen setzt und die Shell mit der Shellbefehlsdatei ausführt, die als offene Datei übergeben wird. Ein Klammerbefehl wird in einer Subshell ausgeführt, ohne die nicht exportierten Bestände zu löschen.

#### Zugehörige Konzepte:

„Verfügbare Shells“ auf Seite 218

Die folgenden Shells werden mit AIX bereitgestellt.

„Koprozesse“ auf Seite 245

In der Korn-Shell oder POSIX-Shell ist es möglich, Befehle als Hintergrundprozesse auszuführen. Diese Befehle, die innerhalb eines Shell-Script ausgeführt werden, werden *Koprozesse* genannt.

„Befehlszeileneditierung in der Korn- bzw. POSIX-Shell“ auf Seite 257

Normalerweise geben Sie eine Befehlszeile an einem Terminal ein und beenden sie mit einem Zeilenvorschubzeichen (**Eingabetaste** oder **Zeilenvorschub**). Wenn Sie die Befehlszeileneditierungsoption *emacs*, *gmacs* oder *vi* aktivieren, können Sie die Befehlszeile editieren.

„Arithmetische Berechnungen in der Korn- bzw. POSIX-Shell“ auf Seite 228

Mit dem integrierten Befehl **let** der Korn- bzw. POSIX-Shell können Sie Ganzzahlberechnungen durchführen.

„Integrierte Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 246

Sonderbefehle sind in die Korn- bzw. POSIX-Shell integriert und werden im Shellprozess ausgeführt.

#### *Zusammengesetzte Befehle in der Korn-Shell:*

Ein zusammengesetzter Befehl kann eine Liste einfacher Befehle oder eine Kette sein oder mit einem reservierten Wort beginnen. Am häufigsten werden Sie zusammengesetzte Befehle wie **if**, **while** und **for** verwenden, wenn Sie Shell-Scripts schreiben.

Im Folgenden finden Sie eine Liste der zusammengesetzten Befehle für die Korn- bzw. POSIX-Shell:

Befehlssyntax	Beschreibung
<b>for</b> <i>ID</i> [ <b>in</b> <i>Wort ...</i> ] ; <b>do</b> <i>Liste</i> ; <b>done</b>	Bei jeder Ausführung des Befehls <b>for</b> wird der Parameter <b>ID</b> auf das nächste Wort aus der Liste <b>in</b> <i>Wort</i> gesetzt. Wird der Befehl <b>in</b> <i>Wort ...</i> nicht angegeben, führt der Befehl <b>for</b> den Befehl <b>do</b> <i>Liste</i> einmal für jeden gesetzten positionsgebundenen Parameter aus. Die Ausführung ist beendet, wenn keine weiteren Wörter in der Liste enthalten sind.
<b>select</b> <i>ID</i> [ <b>in</b> <i>Wort ...</i> ] ; <b>do</b> <i>Liste</i> ; <b>done</b>	Beim Befehl <b>select</b> erfolgt die Ausgabe der angegebenen Wortgruppen in die Standardfehlerdatei (Dateideskriptor2). Jedem Wort wird hierbei eine Zahl vorangestellt. Falls der Befehl <b>in</b> <i>Wort ...</i> ausgelassen wird, werden stattdessen die positionsgebundenen Parameter verwendet. Es wird die Eingabeaufforderung <b>PS3</b> ausgegeben und eine Zeile aus der Standardeingabe gelesen. Wenn diese Zeile die Nummer eines der aufgelisteten Wörter enthält, wird der Wert des Parameters <i>ID</i> auf das Wort gesetzt, das dieser Nummer entspricht.  Falls die aus der Standardeingabe gelesene Zeile leer ist, wird die Auswahlliste erneut ausgegeben. Andernfalls wird der Wert des Parameters <i>ID</i> auf null gesetzt. Der Inhalt der aus der Standardeingabe gelesenen Zeile wird im Parameter <b>ANTWORT</b> gespeichert. Der Parameter <i>Liste</i> wird für jede Auswahl ausgeführt, bis ein Unterbrechungs- oder Dateiendezeichen (eof) gefunden wird.
<b>case</b> <i>Wort in</i> [( <i>(</i> <i>Muster</i> [   <i>Muster</i> ] ... ) <i>Liste</i> ;;] ... <b>esac</b>	Der Befehl <b>case</b> führt den Parameter <i>Liste</i> aus, der dem ersten Parameter <i>Muster</i> zugeordnet ist, der mit dem Parameter <i>Wort</i> übereinstimmt. Das Format der Muster entspricht dem bei der Dateinamensubstitution verwendeten Format.
<b>if</b> <i>Liste</i> ; <b>then</b> <i>Liste</i> [ <b>elif</b> <i>Liste</i> ; <b>then</b> <i>Liste</i> ] ... [ <b>else</b> <i>Liste</i> ] ; <b>fi</b>	Der Parameter <i>Liste</i> gibt eine Liste der auszuführenden Befehle an. Die Shell führt den Befehl <b>if</b> <i>Liste</i> zuerst aus. Wird ein Exit-Status gleich null zurückgegeben, wird der Befehl <b>then</b> <i>Liste</i> ausgeführt. Andernfalls werden die Befehle ausgeführt, die im Parameter <i>Liste</i> angegeben sind, der dem Befehl <b>elif</b> folgt.  Falls der Wert, den der letzte Befehl <b>elif</b> <i>Liste</i> zurückgibt, gleich null ist, wird der Befehl <b>then</b> <i>Liste</i> ausgeführt. Ist der Wert, den der letzte Befehl <b>then</b> <i>Liste</i> zurückgibt, gleich null, wird der Befehl <b>else</b> <i>Liste</i> ausgeführt. Wenn keiner der Befehle, die im Parameter <i>Liste</i> angegeben sind, für die Befehle <b>else</b> bzw. <b>then</b> ausgeführt wird, gibt der Befehl <b>if</b> einen Exit-Status gleich null (0) zurück.
<b>while</b> <i>Liste</i> ; <b>do</b> <i>Liste</i> ; <b>done</b> <b>until</b> <i>Liste</i> ; <b>do</b> <i>Liste</i> ; <b>done</b>	Der Parameter <i>Liste</i> gibt eine Liste der auszuführenden Befehle an. Der Befehl <b>while</b> führt wiederholt Befehle aus, die im Parameter <i>Liste</i> angegeben sind. Wenn der Exit-Status des letzten Befehls im Befehl <b>while</b> <i>Liste</i> gleich null ist, wird der Befehl <b>do</b> <i>Liste</i> ausgeführt. Ist der Exit-Status des letzten Befehls im Befehl <b>while</b> <i>Liste</i> ungleich null, wird die Schleife beendet. Wenn keine Befehle im Befehl <b>do</b> <i>Liste</i> ausgeführt werden, gibt der Befehl <b>while</b> den Exit-Status null zurück. Der Befehl <b>until</b> kann anstelle des Befehls <b>while</b> verwendet werden, um den Schleifenabschluss zu negieren.
( <i>Liste</i> )	Der Parameter <i>Liste</i> gibt eine Liste mit auszuführenden Befehlen an. Die Shell führt den Parameter <i>Liste</i> in einer separaten Umgebung aus. <b>Anmerkung:</b> Sind für eine Verschachtelung zwei benachbarte offene runde Klammern erforderlich, muss dazwischen ein Leerzeichen eingefügt werden, um zwischen dem Befehl und der arithmetischen Berechnung zu unterscheiden.
{ <i>Liste</i> ; }	Der Parameter <i>Liste</i> gibt eine Liste mit auszuführenden Befehlen an. Der Parameter <i>Liste</i> wird einfach ausgeführt.  <b>Anmerkung:</b> Anders als die Metazeichen ( ) kennzeichnen geschweifte Klammern ( { } ) reservierte Wörter. (Reservierte Wörter werden für spezielle Zwecke und nicht als benutzerdeklarierte Kennungen verwendet). Damit diese reservierten Wörter erkannt werden, müssen sie am Anfang einer Zeile oder hinter einem Semikolon (;) stehen.
[[ <i>Ausdruck</i> ]]	Wertet den Parameter <i>Ausdruck</i> aus. Falls der Ausdruck wahr ist, gibt der Befehl einen Exit-Status gleich null zurück.
<b>function</b> <i>ID</i> { <i>Liste</i> ; } oder <b>function</b> <i>ID</i> 0 { <i>Liste</i> ; }	Definiert eine Funktion, auf die mit dem Parameter <i>ID</i> verwiesen wird. Der Hauptteil der Funktion ist die angegebene Liste von Befehlen zwischen den geschweiften Klammern ( { } ). Die Angabe ( ) besteht aus zwei Operatoren. Daher können Leerzeichen mit der <i>ID</i> , ( und ) gemischt werden. Dies ist jedoch nicht erforderlich.
<b>time</b> <i>Befehlskette</i>	Führt den Parameter <i>Befehlskette</i> aus. Die abgelaufene Zeit, die Benutzerzeit und die Systemzeit werden in der Standardfehlerausgabe aufgezeichnet.

## Zugehörige Konzepte:

„Parameter in der Korn-Shell“ auf Seite 239

Im Folgenden finden Sie eine Beschreibung der Korn-Shellparameter.

*Shell starten:*

Sie können die Korn-Shell mit dem Befehl **ksh**, dem Befehl **psh** (POSIX-Shell) oder dem Befehl **exec** starten.

Wenn die Shell mit dem Befehl **exec** gestartet wird und das erste Zeichen des Nullarguments (**\$0**) ein Minuszeichen (-) ist, wird angenommen, dass die Shell eine Anmeldeshell ist. Die Shell liest zuerst die Befehle aus der Datei `/etc/profile` und anschließend die Befehle aus der Datei `.profile` im aktuellen Verzeichnis bzw. aus der Datei `$HOME/.profile` (sofern vorhanden). Danach liest die Shell die Befehle aus der Datei, deren Name durch Parametersubstitution im Wert der Umgebungsvariablen **ENV** ermittelt wurde (sofern vorhanden).

Wenn beim Aufruf der Korn- bzw. POSIX-Shell der Parameter *Datei* [*Parameter*] angegeben wird, führt die Shell die mit dem Parameter *Datei* angegebene Scriptdatei einschließlich aller angegebenen Parameter aus. Die angegebene Scriptdatei muss Leseberechtigung besitzen. Alle **setuid**- und **setgid**-Einstellungen werden ignoriert. Anschließend liest die Shell die Befehle.

**Anmerkung:** Geben Sie beim Aufruf der Korn- bzw. POSIX-Shell mit den Flags **-c** und **-s** keine Scriptdatei an.

Nähere Informationen zu positionsgebundenen Parametern finden Sie im Abschnitt „Parameter in der Korn-Shell“ auf Seite 239.

**Zugehörige Konzepte:**

„Parameter in der Korn-Shell“ auf Seite 239

Im Folgenden finden Sie eine Beschreibung der Korn-Shellparameter.

*Korn-Shellumgebung:*

Alle Variablen (mit den zugeordneten Werten), die einem Befehl zu Beginn der Ausführung bekannt sind, bestimmen dessen *Umgebung*.

Die Umgebung enthält Variablen, die ein Befehl von seinem übergeordneten Prozess erbt, und Variablen, die als Schlüsselwortparameter zum Aufrufen des Befehls in der Befehlszeile angegeben werden. Die Shell kommuniziert auf verschiedene Arten mit der Umgebung. Beim Starten durchsucht die Shell die Umgebung und erstellt einen Parameter für jeden gefundenen Namen, ordnet jedem Parameter einen entsprechenden Wert zu und markiert diesen für den Export. Diese Umgebung wird von den ausgeführten Befehlen übernommen.

Werden die Werte der Shellparameter geändert oder mit dem Befehl **export** oder **typeset -x** neue Werte erstellt, werden diese Parameter Teil der Umgebung. Aus Sicht eines ausgeführten Befehls setzt sich die Umgebung demzufolge aus allen Name/Wert-Paaren zusammen, die ursprünglich von der Shell übernommen wurden, und deren Werte möglicherweise von der aktuellen Shell geändert wurden, sowie aus allen zusätzlichen Parametern, die mit dem Befehl **export** oder **typeset -x** erstellt wurden. Der ausgeführte Befehl (Subshell) erkennt alle Änderungen, die an den übernommenen Umgebungsvariablen vorgenommen wurden. Um die geänderten Werte für ihre untergeordneten Shells und Prozesse zu erkennen, muss die Subshell die Variablen jedoch exportieren.

Die Umgebung für jeden einfachen Befehl oder jede einfache Funktion wird geändert, indem eine Parameterzuordnung vorangestellt wird bzw. mehrere Parameterzuordnungen vorangestellt werden. Ein Argument für eine Parameterzuordnung ist ein Wort im Format *ID=Wert*. Daher sind die folgenden beiden Ausdrücke äquivalent (was die Ausführung des Befehls betrifft).

TERM=450 Befehlsargumente

(export TERM; TERM=450; Befehlsargumente)

### *Funktionen in der Korn-Shell:*

Das reservierte Wort **function** definiert Shellfunktionen. Die Shell liest und speichert Funktionen intern. Aliasnamen werden beim Lesen der Funktion aufgelöst. Funktionen werden von der Shell auf dieselbe Weise ausgeführt wie Befehle, wobei die Argumente als positionsgebundene Parameter übergeben werden.

Die Korn- bzw. POSIX-Shell führt Funktionen in der Umgebung aus, in der die Funktionen aufgerufen wurden. Die folgenden Angaben werden von der Funktion und dem aufrufenden Script gemeinsam benutzt, was zu gewissen Nebenwirkungen führen kann:

- Variablenwerte und Attribute (sofern Sie den Befehl **typeset** nicht in der Funktion verwenden, um eine lokale Variable zu deklarieren)
- Arbeitsverzeichnis
- Aliasnamen, Funktionsdefinitionen und Attribute
- Sonderparameter \$
- Geöffnete Dateien

Folgende Angaben werden von der Funktion und dem aufrufenden Prozess nicht gemeinsam benutzt, wodurch keine Nebenwirkungen erzeugt werden:

- positionsgebundene Parameter,
- Sonderparameter #,
- Variablen in einer Variablenzuordnungsliste beim Aufruf der Funktion,
- in der Funktion mit dem Befehl **typeset** deklarierte Variablen,
- Optionen,
- Traps. Signale, die vom aufrufenden Script ignoriert werden, werden jedoch auch von der Funktion ignoriert.

**Anmerkung:** In früheren Versionen der Korn-Shell wurden Traps mit Ausnahme von **EXIT** und **ERR** von der Funktion und dem aufrufenden Script gemeinsam verwendet.

Wird der Trap für **0** oder **EXIT** im Hauptteil einer Funktion ausgeführt, wird die Aktion nach Beendigung der Funktion in der Umgebung ausgeführt, in der die Funktion aufgerufen wurde. Wird der Trap *außerhalb* des Hauptteils einer Funktion ausgeführt, wird die Aktion beim Verlassen der Korn-Shell ausgeführt. In früheren Versionen der Korn-Shell wurde kein Trap für **0** oder **EXIT** außerhalb des Hauptteils einer Funktion beim Verlassen der Funktion ausgeführt.

Wenn eine Funktion ausgeführt wird, hat sie dieselben Eigenschaften für Syntaxfehler und Variablenzuordnung, die auch für die integrierten Korn- bzw. POSIX-Shellbefehle beschrieben werden.

Der zusammengesetzte Befehl wird ausgeführt, wenn als Funktionsname der Name eines einfachen Befehls angegeben wird. Die Operanden des Befehls werden während der Ausführung des zusammengesetzten Befehls vorübergehend zu positionsgebundenen Parametern. Der Sonderparameter # wird ebenfalls geändert und zeigt die Anzahl der Operanden an. Der Sonderparameter 0 wird nicht geändert.

Der Sonderbefehl **return** wird verwendet, um von Funktionsaufrufen zurückzukehren. Bei Fehlern innerhalb von Funktionen wird die Steuerung an das aufrufende Programm zurückgegeben.

Funktionsangaben können mit der Option **-f** oder **+f** des Sonderbefehls **typeset** aufgelistet werden. Außerdem listet die Option **-f** den Text von Funktionen auf. Die Definition von Funktionen wird mit der Option **-f** des Sonderbefehls **unset** aufgehoben.

Normalerweise werden Funktionen inaktiviert, wenn die Shell ein Shell-Script ausführt. Mit der Option **-xf** des Sonderbefehls **typeset** kann eine Funktion in Scripts exportiert werden, die ohne einen separaten Aufruf der Shell ausgeführt werden. Funktionen, die über einen separaten Aufruf der Shell definiert werden müssen, sollten in der Datei ENV mit der Option **-xf** des Sonderbefehls **typeset** angegeben werden.

Der Exit-Status einer Funktionsdefinition ist gleich null, wenn die Funktion nicht erfolgreich deklariert wurde. Andernfalls ist der Exit-Status größer als null. Der Exit-Status eines Funktionsaufrufs entspricht dem Exit-Status des zuletzt von der Funktion ausgeführten Befehls.

#### **Zugehörige Konzepte:**

„Parameter in der Korn-Shell“ auf Seite 239

Im Folgenden finden Sie eine Beschreibung der Korn-Shellparameter.

„Integrierte Befehle in der Korn- bzw. POSIX-Shell“ auf Seite 246

Sonderbefehle sind in die Korn- bzw. POSIX-Shell integriert und werden im Shellprozess ausgeführt.

#### *Befehlsprotokoll in der Korn- bzw. POSIX-Shell:*

Die Korn- bzw. POSIX-Shell speichert die an einem Terminal eingegebenen Befehle in einer Protokolldatei.

Falls die Variable *HISTFILE* gesetzt ist, ist der Wert der Variablen der Name der Protokolldatei. Wenn die Variable *HISTFILE* nicht gesetzt ist oder nicht geschrieben werden kann, wird die Datei `$HOME/.sh_history` als Protokolldatei verwendet. Ist die Protokolldatei nicht vorhanden und kann die Korn-Shell sie nicht erstellen oder ist die Protokolldatei vorhanden und hat die Korn-Shell keine Berechtigung zum Hinzufügen, verwendet die Korn-Shell eine temporäre Datei als Protokolldatei. Die Shell greift auf Befehle aller interaktiven Shells über dieselbe benannte Protokolldatei mit den entsprechenden Berechtigungen zu.

Standardmäßig speichert die Korn- bzw. POSIX-Shell den Text der letzten 128 Befehle für andere Benutzer als Root und die letzten 512 Befehle für Root. Die Größe der Protokolldatei (angegeben mit der Variablen *HISTSZIE*) ist nicht begrenzt. Es ist jedoch zu beachten, dass eine sehr große Protokolldatei die Initialisierung der Korn-Shell verlangsamen kann.

#### *Befehlsprotokollsubstitution:*

Mit dem integrierten Befehl **fc** können Teile der Protokolldatei aufgelistet oder editiert werden. Wenn Sie einen Teil der Datei zur Bearbeitung bzw. Auflistung auswählen möchten, geben Sie die Nummer oder das erste bzw. die ersten Zeichen des Befehls ein.

Sie können einen einzelnen Befehl oder einen Bereich von Befehlen angeben.

Wenn Sie mit dem integrierten Befehl **fc** kein Editorprogramm als Argument angeben, wird der Editor verwendet, der mit der Variablen *FCEDIT* angegeben ist. Wenn die Variable *FCEDIT* nicht definiert ist, wird die Datei `/usr/bin/ed` verwendet. Die editierten Befehle werden beim Verlassen des Editors ausgegeben und ausgeführt.

Das Minuszeichen (-) für den Editornamen wird verwendet, um die Editierphase zu überspringen und den Befehl erneut auszuführen. In diesem Fall kann ein Substitutionsparameter der Form `Alt=Neu` verwendet werden, um den Befehl vor der Ausführung zu ändern. Ist beispielsweise für *r* der Aliasname **fc** `-e` definiert, wird durch die Eingabe von `r bad=good c` der letzte Befehl ausgeführt, der mit dem Buchstaben *c* beginnt, und beim ersten Auftreten die Zeichenfolge `bad` durch `good` ersetzt.

#### **Zugehörige Tasks:**

„Zuvor eingegebene Befehle auflisten (Befehl history)“ auf Seite 137

Mit dem Befehl **history** können Sie die Befehle auflisten, die Sie zuvor eingegeben haben.

## Befehls-Aliasing in der Korn- bzw. POSIX-Shell:

In der Korn- bzw. POSIX-Shell können Sie Aliasnamen für die Anpassung von Befehlen erstellen.

Der Befehl **alias** definiert ein Wort im Format Name=Zeichenfolge als Aliasnamen. Wird ein Aliasname als erstes Wort in einer Befehlszeile verwendet, prüft die Korn-Shell, ob dieser Aliasname bereits verarbeitet wird. Ist dies der Fall, wird der Aliasname von der Korn-Shell nicht ersetzt. Andernfalls wird der Aliasname von der Korn-Shell durch den Wert des Aliasnamens ersetzt.

Das erste Zeichen eines Aliasnamens kann jedes druckbare Zeichen außer einem Metazeichen sein. Die restlichen Zeichen müssen wie bei einer gültigen Kennung gewählt werden. Die Ersetzungszeichenfolge kann einen beliebigen gültigen Shelltext einschließlich Metazeichen enthalten.

Falls das letzte Zeichen des Aliaswertes ein Leerzeichen ist, prüft die Shell außerdem, ob das dem Aliasnamen folgende Wort für eine Aliassubstitution in Frage kommt. Sie können Aliasnamen verwenden, um integrierte Sonderbefehle erneut zu definieren. Reservierte Wörter können mit Aliasnamen nicht erneut definiert werden. Aliasdefinitionen werden bei Aufrufen der Korn-Shell (**ksh**) nicht geerbt. Wenn Sie jedoch **alias -x** angeben, bleibt der Aliasname für die Scripts wirksam, die im Kontext der aktuellen Shell ausgeführt werden und keine eigene Shell aufrufen. Sie müssen **alias -x** und die Aliasdefinition in Ihrer Umgebungsdatei angeben, wenn Sie eine Aliasdefinition exportieren und dafür sorgen möchten, dass alle Kindprozesse auf diese Definition zugreifen können.

Mit dem Befehl **alias** können Sie Aliasnamen erstellen, auflisten und exportieren.

Mit dem Befehl **unalias** können Sie Aliasnamen entfernen.

Zum Erstellen eines Aliasnamens wird folgendes Format verwendet:

```
alias Name=Zeichenfolge
```

In diesem Beispiel steht der Parameter **Name** für den Namen und der Parameter **Zeichenfolge** für den Wert des Aliasnamens.

Die folgenden exportierten Aliasnamen sind von der Korn-Shell vordefiniert. Sie können jedoch inaktiviert oder neu definiert werden. Es wird empfohlen, diese Aliasnamen nicht zu ändern. Eine Änderung könnte später zu Verwirrungen führen, da diese Aliasnamen dann anders als von der Korn-Shell vordefiniert funktionieren.

```
autoload='typeset -fu'
false='let 0'
functions='typeset -f'
functions='typeset -f'
hash='alias -t'
history='fc -l'
integer='typeset -i'
nohup='nohup '
r='fc -e -'
true=':'
type='whence -v'
```

Aliasnamen werden in nicht interaktiven Aufrufen der Korn-Shell (**ksh**) nicht unterstützt. Beispielsweise werden Aliasnamen in einem Shell-Script oder bei Angabe der Option **-c** in **ksh** nicht unterstützt:

```
ksh -c alias
```

### Zugehörige Tasks:

„Befehlsalias erstellen (Shellbefehl alias)“ auf Seite 140

Mit einem *Aliasnamen* können Sie Namen für den Direktaufruf von Befehlen, Dateien und Shellprozeduren erstellen. Durch die Verwendung von Aliasnamen können Sie Zeit einsparen, wenn bestimmte Tasks häufig ausgeführt werden müssen. Sie können einen Befehlsalias erstellen.

*Aliasnamen mit absoluter Pfadangabe:*

Aliasnamen werden häufig als Kurzform für vollständige Pfadnamen verwendet. Mit einer Option des Hilfsprogramms für die Aliasumsetzung kann der Wert eines Aliasnamens automatisch auf den vollständigen Pfadnamen des entsprechenden Befehls gesetzt werden. Diese besondere Art von Aliasname wird als *Aliasname mit absoluter Pfadangabe* bezeichnet.

Aliasnamen mit absoluter Pfadangabe beschleunigen die Ausführung des Befehls, da die Shell nicht mehr in der Variablen *PATH* nach dem vollständigen Pfadnamen suchen muss.

Der Befehl **set -h** aktiviert die *Befehlsprotokollierung*, so dass die Shell jedes Mal, wenn auf einen Befehl verwiesen wird, den Wert eines Aliasnamens mit absoluter Pfadangabe definiert. Die Definition dieses Wertes wird wieder aufgehoben, wenn die Variable *PATH* zurückgesetzt wird.

Die Aliasnamen bleiben mit der absoluten Pfadangabe erhalten, so dass der Wert durch den nächsten Verweis neu definiert wird. Verschiedene Aliasnamen mit absoluter Pfadangabe sind in der Shell kompiliert.

*Tildensubstitution:*

Nach der Aliassubstitution überprüft die Shell jedes Wort daraufhin, ob es mit einer Tilde (^) beginnt, die nicht in Anführungszeichen steht. Findet die Shell ein solches Wort, überprüft sie das Wort bis zum ersten Schrägstrich (/) daraufhin, ob es mit einem Benutzernamen in der Datei */etc/passwd* übereinstimmt. Wird eine Übereinstimmung gefunden, ersetzt die Shell die Tilde (^) und den Namen durch das Anmeldeverzeichnis des übereinstimmenden Benutzers. Dieser Prozess wird als *Tildensubstitution* bezeichnet.

Findet die Shell keine Übereinstimmung, wird der ursprüngliche Text nicht geändert. Außerdem führt die Korn-Shell besondere Ersetzungen durch, wenn die Tilde (^) das einzige Zeichen im Wort ist oder vor einem Pluszeichen (+) oder Minuszeichen (-) steht:

Eintrag	Beschreibung
~	Wird durch den Wert der Variablen <i>HOME</i> ersetzt.
~+	Wird durch die Variable <i>\$PWD</i> (den vollständigen Pfadnamen des aktuellen Verzeichnisses) ersetzt.
~-	Wird durch die Variable <i>\$OLDPWD</i> (den vollständigen Pfadnamen des vorherigen Verzeichnisses) ersetzt.

Zusätzlich versucht die Shell, die Tildensubstitution durchzuführen, wenn der Wert eines Variablenzuordnungsparameters mit einer Tilde (^) beginnt.

## Bourne-Shell

Die Bourne-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlsorientierte Programmiersprache.

Der Befehl **bsh** führt die Bourne-Shell aus.

Die Bourne-Shell kann entweder als Anmeldeshell oder als Subshell innerhalb der Anmeldeshell ausgeführt werden. Die Bourne-Shell kann nur mit dem Befehl **login** als Anmeldeshell aufgerufen werden. Dies erfolgt mithilfe des Befehls **bsh** in einem besonderen Format: **-bsh**. Beim Aufruf mit einem dem Befehl vorangehenden Minuszeichen (-) liest die Shell erst Befehle in der Systemdatei */etc/profile* und in der Datei *\$HOME/.profile*, falls vorhanden, und führt diese Befehle aus. Die Datei */etc/profile* legt Variablen fest, die von allen Benutzern benötigt werden. Danach ist die Shell bereit, Befehle von der Standardeingabe des Benutzers zu lesen.

Wird der Parameter **Datei** [*Parameter*] beim Aufruf der Bourne-Shell angegeben, führt sie die im Parameter **Datei** angegebene Scriptdatei mit allen angegebenen Parametern aus. Die angegebene Scriptdatei muss Leseberechtigung besitzen. Alle *setuid*- und *setgid*-Einstellungen werden ignoriert. Anschließend liest die Shell die Befehle. Geben Sie kein Script an, wenn Sie das Flag **-c** oder das Flag **-s** angeben.

**Zugehörige Konzepte:**

„Verfügbare Shells“ auf Seite 218

Die folgenden Shells werden mit AIX bereitgestellt.

### Bourne-Shellumgebung:

Alle Variablen (mit den zugeordneten Werten), die einem Befehl zu Beginn der Ausführung bekannt sind, bestimmen dessen *Umgebung*. Die Umgebung enthält Variablen, die ein Befehl von seinem Elternprozess übernimmt, und Variablen, die als Schlüsselwortparameter zum Aufrufen des Befehls in der Befehlszeile angegeben werden.

Die Shell übergibt den Kindprozessen die Variablen als Argumente für den integrierten Befehl **export**. Dieser Befehl setzt die benannten Variablen in der Umgebung der Shell und aller zukünftigen Kindprozesse.

Schlüsselwortparameter sind Variable/Wert-Paare, die in der Form von Zuordnungen in der Regel vor dem Prozedurnamen in einer Befehlszeile vorkommen (siehe auch das Flag für den Befehl **set**). Diese Variablen werden in der Umgebung der aufzurufenden Prozedur gesetzt.

Beispiele:

- Beispielsweise kann die folgende Prozedur verwendet werden, um Werte von zwei Variablen (die in einer Befehlsdatei mit dem Namen `key_command` gespeichert sind) anzuzeigen:

```
key_command
echo $a $b
```

Diese Befehlszeilen erzeugen folgende Ausgabe:

Input	Output
a=key1 b=key2 key_command	key1 key2
a=tom b=john key_command	tom john

Die Schlüsselwortparameter einer Prozedur sind nicht im Parameterzähler enthalten, der im Parameter `$#` gespeichert wird.

Eine Prozedur kann auf die Werte aller Variablen in ihrer Umgebung zugreifen. Wenn sie einen dieser Werte ändert, werden diese Änderungen jedoch nicht in der Shellumgebung nachvollzogen. Die Änderungen gelten nur lokal für die betreffende Prozedur. Um die Änderungen in der Umgebung nachzuvollziehen, die die Prozedur an seine Kindprozesse übergibt, müssen die neuen Werte innerhalb dieser Prozedur exportiert werden.

Beispiele:

- Geben Sie Folgendes ein, um eine Liste der Variablen abzurufen, die von der aktuellen Shell exportiert werden können:

```
export
```

- Geben Sie Folgendes ein, um eine Liste der schreibgeschützten Variablen in der aktuellen Shell abzurufen:

```
readonly
```

- Geben Sie Folgendes ein, um eine Liste der Variable/Wert-Paare in der aktuellen Umgebung abzurufen:

```
env
```

Nähere Informationen zu Benutzerumgebungen finden Sie im Abschnitt „Datei `/etc/environment`“ auf Seite 333.



## Bedingte Substitution in der Bourne-Shell:

Normalerweise ersetzt die Shell den Ausdruck `$Variable` durch den Zeichenfolgewart, der der Variablen `Variable`, falls vorhanden, zugeordnet ist. Es gibt jedoch eine spezielle Schreibweise, mit der eine *bedingte Substitution* ausgeführt werden kann, abhängig davon, ob die Variable gesetzt ist und/oder nicht leer ist.

Per Definition ist eine Variable gesetzt, wenn ihr zu einem beliebigen Zeitpunkt ein Wert zugeordnet wurde. Der Wert einer Variablen kann eine leere Zeichenfolge sein, die einer Variablen auf eine der folgenden Arten zugeordnet werden kann:

Eintrag	Beschreibung
<code>A=</code>	
<code>bcd=""</code>	
<code>Efg= ''</code>	Ordnet A, bcd und Efg die leere Zeichenfolge zu.
<code>set '' ''</code>	Setzt den ersten und zweiten positionsgebundenen Parameter auf die leere Zeichenfolge und inaktiviert alle anderen positionsgebundenen Parameter.

Die folgende Liste enthält die verfügbaren Ausdrücke, die zur Ausführung der bedingten Substitution verwendet werden können:

Eintrag	Beschreibung
<code>\${Variable- Zeichenfolge}</code>	Ist die Variable gesetzt, wird der Variablenwert für den Ausdruck eingesetzt. Andernfalls wird der Ausdruck durch <i>Zeichenfolge</i> ersetzt.
<code>\${Variable:-Zeichenfolge}</code>	Ist die Variable gesetzt und ungleich null, wird der Variablenwert für diesen Ausdruck eingesetzt. Andernfalls wird der Ausdruck durch <i>Zeichenfolge</i> ersetzt.
<code>\${Variable=Zeichenfolge}</code>	Ist die Variable gesetzt, wird der Variablenwert für den Ausdruck eingesetzt. Andernfalls wird <i>Variable</i> auf <i>Zeichenfolge</i> gesetzt und anschließend der Variablenwert für diesen Ausdruck eingesetzt. Positionsgebundene Parameter können auf diese Art keine Werte zugeordnet werden.
<code>\${Variable:=Zeichenfolge}</code>	Ist die Variable gesetzt und nicht null, wird der Variablenwert für diesen Ausdruck eingesetzt. Andernfalls wird <i>Variable</i> auf <i>Zeichenfolge</i> gesetzt und anschließend der Variablenwert für diesen Ausdruck eingesetzt. Positionsgebundene Parameter können auf diese Art keine Werte zugeordnet werden.
<code>\${Variable?Zeichenfolge}</code>	Ist die Variable gesetzt, wird der Variablenwert für den Ausdruck eingesetzt. Andernfalls wird eine Nachricht im folgenden Format angezeigt: <code>Variable: Zeichenfolge</code>  Die aktuelle Shell wird beendet (sofern es sich nicht um die Anmeldeshell handelt). Ist kein Wert für die Variable <i>Zeichenfolge</i> angegeben, zeigt die Shell folgende Nachricht an: <code>Variable: Parameter null oder nicht gesetzt</code>
<code>\${Variable?:Zeichenfolge}</code>	Ist die Variable gesetzt und ungleich null, wird der Variablenwert für diesen Ausdruck eingesetzt. Andernfalls wird eine Nachricht im folgenden Format angezeigt: <code>Variable: Zeichenfolge</code>  Die aktuelle Shell wird beendet (sofern es sich nicht um die Anmeldeshell handelt). Wird <i>Zeichenfolge</i> nicht angegeben, zeigt die Shell folgende Nachricht an: <code>Variable: Parameter null oder nicht gesetzt</code>
<code>\${Variable+Zeichenfolge}</code>	Ist die Variable gesetzt, wird <i>Zeichenfolge</i> für diesen Ausdruck eingesetzt. Andernfalls wird die leere Zeichenfolge ersetzt.
<code>\${Variable:+Zeichenfolge}</code>	Ist die Variable gesetzt und ungleich null, wird <i>Zeichenfolge</i> für diesen Ausdruck eingesetzt. Andernfalls wird die leere Zeichenfolge ersetzt.

Bei der bedingten Substitution wertet die Shell die Variable *Zeichenfolge* erst aus, wenn die Shell diese Variable als Ersatzzeichenfolge verwendet. Demzufolge führt die Shell im folgenden Beispiel den Befehl `pwd` nur aus, wenn `d` nicht gesetzt oder leer ist.

```
echo ${d:-`pwd`}
```

### **Zugehörige Konzepte:**

„Benutzerdefinierte Variablen in der Bourne-Shell“ auf Seite 284

Die Bourne-Shell erkennt alphanumerische Variablen, denen Zeichenfolgewerte zugeordnet werden können.

### **Positionsgebundene Parameter in der Bourne-Shell:**

Wird eine Shellprozedur ausgeführt, erstellt die Shell implizit positionsgebundene Parameter, die auf jedes Wort in der Befehlszeile über seine dortige Position verweisen.

Das Wort an der Position 0 (Prozedurname) wird als \$0 bezeichnet, das nächste Wort (der erste Parameter) als \$1 usw. bis zu \$9. Um auf Befehlszeilenparameter mit einer höheren Nummer als 9 zu verweisen, muss der integrierte Befehl **shift** verwendet werden.

Die Werte der positionsgebundenen Parameter können mit dem integrierten Befehl **set** explizit zurückgesetzt werden.

**Anmerkung:** Wird ein Argument für eine Position nicht angegeben, wird der entsprechende positionsgebundene Parameter auf null gesetzt. Positionsgebundene Parameter sind global und können an verschachtelte Shellprozeduren übergeben werden.

### **Zugehörige Konzepte:**

„Benutzerdefinierte Variablen in der Bourne-Shell“ auf Seite 284

Die Bourne-Shell erkennt alphanumerische Variablen, denen Zeichenfolgewerte zugeordnet werden können.

### **Zugehörige Verweise:**

„Vordefinierte Sondervariablen in der Bourne-Shell“ auf Seite 287

Verschiedene Variablen haben eine Sonderbedeutung. Die folgenden Variablen werden nur von der Bourne-Shell gesetzt:

### **Dateinamenssubstitution in der Bourne-Shell:**

In der Bourne-Shell können Dateinamenssubstitutionen durchgeführt werden.

Befehlsparameter sind häufig Dateinamen. Sie können eine Liste mit Dateinamen automatisch als Parameter in der Befehlszeile erzeugen. Geben Sie dazu ein Zeichen an, das die Shell als Platzhalterzeichen erkennt. Wenn ein Befehl ein solches Zeichen enthält, ersetzt die Shell es durch die Dateinamen in einem Verzeichnis.

**Anmerkung:** Die Bourne-Shell unterstützt keine auf Äquivalenzklassifikation von Zeichen basierende Dateinamenerweiterung.

Die meisten Zeichen in solchen Mustern entsprechen sich selbst, aber es können auch einige besondere Platzhalterzeichen im Muster verwendet werden. Im Folgenden sind diese Sonderzeichen aufgelistet:

Eintrag	Beschreibung
*	Gleicht jede Zeichenfolge ab, einschließlich der leeren Zeichenfolge.
?	Entspricht jedem Einzelzeichen.
[ . . . ]	Entspricht einem der in den eckigen Klammern eingeschlossenen Zeichen.
![ . . . ]	Entspricht einem der in den eckigen Klammern eingeschlossenen Zeichen <i>mit Ausnahme</i> der Zeichen, die dem Ausrufezeichen folgen.

Ein in eckige Klammern eingeschlossenes und durch ein Minuszeichen (-) getrenntes Zeichenpaar gibt alle Zeichen an, die aufgrund ihrer Position innerhalb der aktuellen Sortierfolge zwischen diesen beiden Zeichen liegen.

Die Mustererkennung unterliegt folgenden Einschränkungen: Ist das erste Zeichen eines Dateinamens ein Punkt (.), kann er nur über ein Muster abgeglichen werden, das auch mit einem Punkt beginnt. Beispielsweise stimmt \* (Stern) mit den Dateinamen *meinedatei* und *ihredatei* überein, aber nicht mit den Dateinamen *.meinedatei* und *.ihredatei*. Um diese Dateien zu finden, müssen Sie ein Muster wie das folgende verwenden:

```
.*datei
```

Stimmt das Muster mit keinem Dateinamen überein, wird das Muster selbst als Ergebnis des versuchten Abgleichs zurückgegeben.

Datei- und Verzeichnisnamen dürfen die Zeichen \*, ?, [ und ] nicht enthalten, da dadurch eine endlose Rekursion (d. h. eine Endlosschleife) während der Abgleichversuche auftreten würde.

### Eingabe- und Ausgabeumleitung in der Bourne-Shell:

Es gibt Umleitungsoptionen, die in Befehlen verwendet werden können.

Im Allgemeinen ist es bei den meisten Befehlen unerheblich, ob ihre Eingabe oder Ausgabe der Tastatur, dem Bildschirm oder einer Datei zugeordnet ist. Deshalb kann ein Befehl ohne Schwierigkeiten entweder über die Tastatur eingegeben oder in einer Befehlskette verwendet werden.

Die folgenden Umleitungsoptionen können überall in einem einfachen Befehl auftreten. Sie können auch vor oder nach einem Befehl stehen, werden aber nicht an den Befehl übergeben:

Eintrag	Beschreibung
<Datei	Verwendet die angegebene Datei als Standardeingabe.
>Datei	Verwendet die angegebene Datei als Standardausgabe. Erstellt die Datei, wenn sie nicht existiert. Andernfalls wird sie auf die Länge 0 abgeschnitten.
> >Datei	Verwendet die angegebene Datei als Standardausgabe. Andernfalls wird die Ausgabe am Ende der Datei hinzugefügt.
<<[-]eofstr	Liest als Standardeingabe alle Zeilen aus der Variablen <i>eofstr</i> bis zu der Zeile, die nur <i>eofstr</i> oder ein Dateieindezeichen enthält. Ist ein beliebiges Zeichen in der Variablen <i>eofstr</i> in Anführungszeichen gesetzt, werden die Zeichen in den Eingabezeilen von der Shell nicht ersetzt oder interpretiert. Andernfalls führt die Shell die Variablen- und Befehlssubstitution aus und ignoriert Zeilenvorschubzeichen ( <b>\newline</b> ), die in Anführungszeichen gesetzt sind. Verwenden Sie als Anführungszeichen für Zeichen in der Variablen <i>eofstr</i> oder in den Eingabezeilen einen umgekehrten Schrägstrich ( <b>\</b> ).  Wenn Sie der Umleitungsoption << ein Minuszeichen (-) hinzufügen, werden alle führenden Tabulatorzeichen aus der Variablen <i>eofstr</i> und aus den Eingabezeilen entfernt.
<&Ziffer	Ordnet die Standardeingabe dem in der Variablen <i>Ziffer</i> angegebenen Dateideskriptor zu.
>&Ziffer	Ordnet die Standardausgabe dem in der Variablen <i>Ziffer</i> angegebenen Dateideskriptor zu.
<&-	Schließt die Standardeingabe.
>&-	Schließt die Standardausgabe.

**Anmerkung:** In der Restricted Shell ist die Ausgabeumleitung nicht zulässig.

Weitere Informationen zur Umleitung finden Sie im Abschnitt „Eingabe- und Ausgabeumleitung“ auf Seite 362.

### Liste der integrierten Bourne-Shellbefehle:

Im Folgenden sind die integrierten Bourne-Shellbefehle aufgelistet.

Eintrag	Beschreibung
:	Gibt den Exit-Wert null zurück.
.	Liest und führt Befehle von einem Dateiparameter aus und kehrt dann zurück.
<b>break</b>	Führt mit der Ausführung nach dem nächsten Ende einer <b>for</b> -, <b>while</b> - oder <b>until</b> -Schleife fort.
<b>cd</b>	Wechselt vom aktuellen Verzeichnis in das angegebene Verzeichnis.
<b>continue</b>	Nimmt die nächste Iteration der Befehlsschleife <b>for</b> , <b>while</b> und <b>until</b> auf.
<b>echo</b>	Schreibt Zeichenfolgen in die Standardausgabe.
<b>eval</b>	Liest die Argumente als Eingabe für die Shell und führt die entsprechenden Befehle aus.
<b>exec</b>	Führt an Stelle der Shell den im Parameter <b>Argument</b> angegebenen Befehl aus, ohne einen neuen Prozess zu erstellen.
<b>exit</b>	Verlässt die Shell mit dem im Parameter <b>n</b> angegebenen Exit-Status.
<b>export</b>	Markiert Namen für den automatischen Export in die Umgebung von nachfolgend ausgeführten Befehlen.
<b>hash</b>	Sucht und speichert die Position im Suchpfad von angegebenen Befehlen.
<b>pwd</b>	Zeigt das aktuelle Verzeichnis an.
<b>read</b>	Liest eine Zeile von der Standardeingabe.
<b>readonly</b>	Markiert den im Parameter <b>Name</b> angegebenen Namen als schreibgeschützt.
<b>return</b>	Bewirkt, dass eine Funktion bei einem angegebenen Rückgabewert endet.
<b>set</b>	Steuert die Anzeige verschiedener Parameter für die Standardausgabe.
<b>shift</b>	Verschiebt Argumente der Befehlszeile nach links.
<b>test</b>	Wertet Bedingungsausdrücke aus.
<b>times</b>	Zeigt die aufgelaufenen Benutzer- und Systemzeiten für Prozesse an, die von der Shell ausgeführt wurden.
<b>trap</b>	Führt einen angegebenen Befehl aus, wenn die Shell ein angegebenes oder mehrere angegebene Signal(e) empfängt.
<b>type</b>	Interpretiert, wie die Shell einen angegebenen Namen als Befehlsnamen interpretieren würde.
<b>ulimit</b>	Zeigt zugeordnete Shellressourcen an oder passt diese an.
<b>umask</b>	Legt die Dateiberechtigungen fest.
<b>unset</b>	Entfernt die Variable oder Funktion, die einem angegebenen Namen entspricht.
<b>wait</b>	Wartet darauf, dass der angegebene Kindprozess endet, und berichtet dessen Beendigungsstatus.

### Zugehörige Verweise:

„Integrierte Befehle in der Bourne-Shell“ auf Seite 279

Sonderbefehle sind in die Bourne-Shell integriert und werden im Shellprozess ausgeführt.

### Bourne-Shellbefehle:

Sie können in der Bourne-Shell Befehle absetzen.

Wird in der Bourne-Shell ein Befehl eingegeben, wertet die Shell den Befehl erst aus und führt alle angegebenen Substitutionen durch. Anschließend wird der Befehl ausgeführt, vorausgesetzt, dass

- der Befehlsname ein integrierter Sonderbefehl der Bourne-Shell ist  
ODER
- der Befehlsname mit dem Namen einer definierten Funktion übereinstimmt. In diesem Fall setzt die Shell die positionsgebundenen Parameter auf die Parameter der Funktion.

Stimmt der Befehlsname weder mit einem integrierten Befehl noch mit dem Namen einer definierten Funktion überein, benennt der Befehl aber eine ausführbare Datei, bei der es sich um ein kompiliertes (binäres) Programm handelt, erzeugt die Shell (*als Elternprozess*) einen neuen (*Kind*-) Prozess, der das Programm sofort ausführt. Wenn die Datei als ausführbar markiert, aber kein kompiliertes Programm ist,

behandelt die Shell sie wie eine Shellprozedur. In diesem Fall erzeugt die Shell eine weitere Instanz von sich selbst (eine *Subshell*), um die Datei zu lesen und die darin enthaltenen Befehle auszuführen. Die Shell führt in einer Subshell auch Klammerbefehle aus. Für den Benutzer sieht es so aus, als ob ein kompiliertes Programm genauso ausgeführt wird wie eine Shellprozedur. Die Shell sucht normalerweise in Dateisystemverzeichnissen nach Befehlen. Dabei wird die folgende Reihenfolge eingehalten:

1. /usr/bin
2. /etc
3. /usr/sbin
4. /usr/ucb
5. \$HOME/bin
6. /usr/bin/X11
7. /sbin
8. Aktuelles Verzeichnis

Die Shell durchsucht nacheinander alle Verzeichnisse, bis der gesuchte Befehl gefunden ist.

**Anmerkung:** Die Variable *PATH* legt die Reihenfolge fest, in der die Shell die Verzeichnisse durchsucht. Die Reihenfolge der zu durchsuchenden Verzeichnisse kann durch Ändern der Variablen *PATH* geändert werden.

Wird beim Aufruf eines Befehls ein bestimmter Pfadname angegeben (z. B. /usr/bin/sort), sucht die Shell nur in diesem angegebenen Verzeichnis. Falls der Befehlsname einen Schrägstrich (/) enthält, verwendet die Shell den Suchpfad nicht.

Es ist möglich, einen vollständigen Pfadnamen anzugeben, der mit dem Stammverzeichnis beginnt (z. B. /usr/bin/sort). Der Benutzer kann auch einen Pfadnamen angeben, der auf das aktuelle Verzeichnis verweist. Wenn Sie beispielsweise

```
bin/meinedatei
```

angeben, sucht die Shell im aktuellen Verzeichnis nach dem Verzeichnis bin und in diesem Verzeichnis nach der Datei meinedatei.

**Anmerkung:** Die Restricted Shell führt keine Befehle aus, die einen Schrägstrich (/) enthalten.

Die Shell speichert die Position im Suchpfad jedes ausgeführten Befehls (um später unnötige **exec**-Befehle zu vermeiden). Findet die Shell den Befehl in einem relativen Verzeichnis (einem Verzeichnis, dessen Name nicht mit einem Schrägstrich (/) beginnt), muss die Shell die Position eines Befehls jedes Mal, wenn das aktuelle Verzeichnis gewechselt wird, erneut festlegen. Wenn die Variable *PATH* geändert oder der Befehl **hash -r** ausgeführt wird, verwirft die Shell alle gespeicherten Befehlspositionen.

*Zeichen in Anführungszeichen setzen:*

Viele Zeichen haben für die Shell eine Sonderbedeutung. Manchmal ist es allerdings erforderlich, diese Bedeutung außer Kraft zu setzen. Mit einfachen Anführungszeichen (') und doppelten Anführungszeichen ("), in die Zeichenfolgen eingeschlossen werden, oder einem Backslash (\) vor einem einzelnen Zeichen können diese Zeichen ohne ihre Sonderbedeutung verwendet werden.

Alle Zeichen außer den einschließenden einfachen Anführungszeichen werden ohne Sonderbedeutung interpretiert. Demzufolge ordnet der Befehl

```
stuff='echo $? $*; ls * | wc'
```

die Literalzeichenfolge `echo $? $*; ls * | wc` der Variablen `stuff` zu. Die Shell führt weder die Befehle **echo**, **ls** und **wc** aus, noch erweitert sie die Variablen `?` und `$*` oder das Sonderzeichen `*`.

Wenn das Dollarzeichen (\$), das umgekehrte Anführungszeichen (``) oder das doppelte Anführungszeichen (") von doppelten Anführungszeichen eingeschlossen ist, bleibt die Sonderbedeutung der Zeichen wirksam, während alle anderen Zeichen als Literal übernommen werden. Daher wird bei der Verwendung von doppelten Anführungszeichen die Befehls- und Variablensubstitution ausgeführt. Außerdem betreffen Anführungszeichen keine Befehle innerhalb einer Befehlssubstitution, die Teil der in Anführungszeichen gesetzten Zeichenfolge sind, so dass die Zeichen ihre Sonderbedeutung behalten.

Nehmen Sie z. B. die Folge:

```
ls *
Datei1 Datei2 Datei3
message="Dieses Verzeichnis enthält `ls * ` "
echo $message
Dieses Verzeichnis enthält Datei1 Datei2 Datei3
```

Dies zeigt, dass das Sonderzeichen (\*) (Stern) in der Befehlssubstitution ersetzt wurde.

Wenn die Sonderbedeutung des in doppelte Anführungszeichen eingeschlossenen Dollarzeichens (\$), umgekehrten Anführungszeichens (``) oder doppelten Anführungszeichens (") ignoriert werden soll, setzen Sie vor diese Zeichen einen Backslash (\). Werden keine Anführungszeichen verwendet, hat das Voranstellen des Backslash dieselbe Bedeutung, als wenn das Zeichen in einfache Anführungszeichen gesetzt wird. Ein umgekehrter Schrägstrich direkt vor einem Zeilenvorschubzeichen (d. h. ein umgekehrter Schrägstrich am Ende der Zeile) hebt daher das Zeilenvorschubzeichen auf, und die Befehlszeile kann in der nächsten physischen Zeile fortgesetzt werden.

*Signalverarbeitung:*

Die Shell ignoriert die Signale **INTERRUPT** und **QUIT** für einen aufgerufenen Befehl, wenn der Befehl mit einem Et-Zeichen (&) abgeschlossen wurde, d. h., wenn der Befehl im Hintergrund ausgeführt wird. Andernfalls haben die Signale die Werte, die die Shell von der übergeordneten Shell geerbt hat. Eine Ausnahme ist das Signal **SEGMENTATION VIOLATION**.

Nähere Informationen hierzu finden Sie in der Beschreibung des integrierten Bourne-Shellbefehls **trap**.

*Zusammengesetzte Befehle in der Bourne-Shell:*

Die zusammengesetzten Befehle sind im Folgenden aufgeführt.

- Befehlskette (mindestens ein einfacher Befehl; mehrere einfache Befehle werden durch das Pipe-Symbol (|) voneinander getrennt)
- Liste mit einfachen Befehlen
- Befehl, der mit einem reservierten Wort beginnt
- Befehl der mit dem Steueroperator ( (linke runde Klammer) beginnt

Sofern nicht anders angegeben, ist der von einem zusammengesetzten Befehl zurückgegebene Wert der Wert des zuletzt ausgeführten einfachen Befehls.

*Reservierte Wörter:*

Die folgenden reservierten Wörter für die Bourne-Shell werden nur erkannt, wenn sie ohne Anführungszeichen als erstes Wort eines Befehls angegeben werden.

```
for do done
case esac
if then fi
elif else
while until
{ }
()
```

Eintrag	Beschreibung
<b>for</b> <i>ID</i> [ <b>in</b> <i>Wort</i> . . . ] <b>do</b> <i>Liste</i> <b>done</b>	Setzt den Parameter <i>Kennung</i> auf das Wort bzw. die Wörter, die im Parameter <i>Wort</i> angegeben sind (jeweils einen zur Zeit) und führt die im Parameter <i>Liste</i> angegebenen Befehle aus. Wenn Sie <b>in</b> <i>Wort</i> . . . weglassen, führt der Befehl <b>for</b> den Parameter <i>Liste</i> für jeden gesetzten positionsgebundenen Parameter aus. Die Verarbeitung wird beendet, wenn alle positionsgebundenen Parameter verwendet wurden.
<b>case</b> <i>Wort</i> <b>in</b> <i>Muster</i> [   <i>Muster</i> ] . . . ) <i>Liste</i> ;; [ <i>Muster</i> [   <i>Muster</i> ] . . . ) <i>Liste</i> ;;] . . . <b>esac</b>	Führt die im Parameter <i>Liste</i> angegebenen Befehle aus, die dem ersten Parameter <i>Muster</i> zugeordnet sind, der mit dem Wert des Parameters <i>Wort</i> übereinstimmt. Verwendet dieselbe Schreibweise für Zeichenabgleichung in Mustern, die auch für die Dateinamensubstitution verwendet wird. Die einzige Ausnahme ist, dass ein Schrägstrich (/), ein führender Punkt (.) oder ein Punkt mit Schrägstrich (/.) keine exakte Übereinstimmung erfordern.
<b>if</b> <i>Liste</i> <b>then</b> <i>Liste</i> [ <b>elif</b> <i>Liste</i> <b>then</b> <i>Liste</i> ] . . . [ <b>else</b> <i>Liste</i> ] <b>fi</b>	Führt die im Parameter <i>Liste</i> angegebenen Befehle aus, die dem Befehl <b>if</b> folgen. Gibt der Befehl den Rückkehrcode 0 zurück, führt die Shell den Parameter <i>Liste</i> aus, der dem ersten Befehl <b>then</b> folgt. Andernfalls führt sie den Parameter <i>Liste</i> aus, der dem Befehl <b>elif</b> , falls vorhanden, folgt. Ist dieser Rückkehrcode null, führt die Shell den nächsten Befehl <b>then</b> aus. Schlägt dessen Ausführung fehl, wird der Parameter <i>Liste</i> ausgeführt, der dem Befehl <b>else</b> folgt. Werden keine Befehle <b>else</b> <i>Liste</i> oder <b>then</b> <i>Liste</i> ausgeführt, gibt der Befehl <b>if</b> den Rückkehrcode null zurück.
<b>while</b> <i>Liste</i> <b>do</b> <i>Liste</i> <b>done</b>	Führt die im Parameter <i>Liste</i> angegebenen Befehle aus, die dem Befehl <b>while</b> folgen. Ist der Rückkehrcode des letzten Befehls in <b>while</b> <i>Liste</i> gleich null, führt die Shell den Parameter <i>Liste</i> aus, der dem ersten Befehl <b>do</b> folgt. Die Shell fährt mit dieser Schleife durch die Listen fort, bis der Rückkehrcode des letzten Befehls in <b>while</b> <i>Liste</i> ungleich null ist. Werden keine Befehle für den Befehl <b>do</b> <i>Liste</i> ausgeführt, gibt der Befehl <b>while</b> den Rückkehrcode null zurück.
<b>until</b> <i>Liste</i> <b>do</b> <i>Liste</i> <b>done</b>	Führt die im Parameter <i>Liste</i> angegebenen Befehle aus, die dem Befehl <b>until</b> folgen. Wenn der Exit-Wert des letzten Befehls in <b>until</b> <i>Liste</i> ungleich null ist, wird die <i>Liste</i> ausgeführt, die dem Befehl <b>do</b> folgt. Die Shell setzt die Iteration durch die Listen fort, bis der Rückkehrcode des letzten Befehls im Parameter <b>until</b> <i>Liste</i> null ist. Werden keine Befehle für den Befehl <b>do</b> <i>Liste</i> ausgeführt, gibt der Befehl <b>until</b> den Rückkehrcode null zurück.
( <i>Liste</i> )	Führt die Befehle im Parameter <i>Liste</i> in einer Subshell aus.
{ <i>Liste</i> ; }	Führt die Befehle im Parameter <i>Liste</i> im aktuellen Shellprozess aus und startet keine Subshell.
<i>Name</i> 0 { <i>Liste</i> }	Definiert eine Funktion, auf die mit dem Parameter <i>Name</i> verwiesen wird. Der Hauptteil der Funktion ist die Liste der Befehle zwischen den geschweiften Klammern, die mit dem Parameter <i>Liste</i> angegeben wird.

### Integrierte Befehle in der Bourne-Shell:

Sonderbefehle sind in die Bourne-Shell integriert und werden im Shellprozess ausgeführt.

Sofern nicht anders angegeben, wird die Ausgabe in den Dateideskriptor 1 (Standardausgabe) geschrieben. Der Exit-Status ist 0 (null), wenn der Befehl keine Syntaxfehler enthält. Die Umleitung von Eingabe und Ausgabe ist erlaubt.

Die folgenden Sonderbefehle werden etwas anders verarbeitet als andere Sonderbefehle:

```

: (Doppelpunkt) exec shift
. (Punkt) exit times
break export trap
continue readonly wait
eval return

```

Diese Befehle werden in der Bourne-Shell wie folgt verarbeitet:

- Parameterzuordnungslisten, die dem Befehl vorangehen, bleiben auch nach Abschluss des Befehls gültig.
- Umleitungen von Eingabe und Ausgabe werden nach der Parameterzuordnung verarbeitet.
- Fehler in einem Shell-Script führen dazu, dass die Verarbeitung des Scripts gestoppt wird.

**Zugehörige Verweise:**

„Liste der integrierten Bourne-Shellbefehle“ auf Seite 276  
 Im Folgenden sind die integrierten Bourne-Shellbefehle aufgelistet.

Beschreibungen der Sonderbefehle:

Die Bourne-Shell stellt die folgenden integrierten Sonderbefehle zur Verfügung.

Eintrag	Beschreibung
:	Gibt den Rückkehrcode null zurück.
. <i>Datei</i>	Liest die Befehle aus der angegebenen <i>Datei</i> , führt sie aus und kehrt zurück. Es wird keine Subshell gestartet. Die Shell verwendet den mit der Variablen <i>PATH</i> angegebenen Suchpfad, um das Verzeichnis mit der angegebenen <i>Datei</i> zu finden.
<b>break</b> [ <i>n</i> ]	Führt mit der Ausführung nach dem nächsten Ende einer <b>for</b> -, <b>while</b> - oder <b>until</b> -Schleife fort. Wenn Sie die Variable <i>n</i> angeben, fährt der Befehl <b>break</b> mit der Ausführung nach dem <i>n</i> -ten Schleifenende fort.
<b>continue</b> [ <i>n</i> ]	Nimmt die nächste Iteration der Befehlsschleife <b>for</b> , <b>while</b> und <b>until</b> auf. Wenn Sie die Variable <i>n</i> angeben, setzt der Befehl die Ausführung bei der <i>n</i> -ten Befehlsschleife fort.
<b>cd</b> <i>Verzeichnis</i> ]	Wechselt vom aktuellen Verzeichnis in das angegebene <i>Verzeichnis</i> . Ist kein <i>Verzeichnis</i> angegeben, wird der Wert der Shellvariablen <i>HOME</i> verwendet. Die Shellvariable <i>CDPATH</i> definiert den Suchpfad für das angegebene <i>Verzeichnis</i> . <i>CDPATH</i> ist eine Liste mit durch Doppelpunkte voneinander getrennten Verzeichnisnamen. Ein leerer Pfadname gibt das aktuelle Verzeichnis an (das der Standardpfad ist). Der leere Pfadname wird unmittelbar nach dem Gleichheitszeichen in der Zuordnung oder zwischen begrenzenden Doppelpunkten an einer beliebigen Stelle in der Pfadliste aufgeführt. Beginnt das angegebene <i>Verzeichnis</i> mit einem Schrägstrich (/), verwendet die Shell den Suchpfad nicht. Andernfalls durchsucht die Shell jedes Verzeichnis in der Shellvariablen <i>CDPATH</i> .  <b>Anmerkung:</b> In der Restricted Shell kann der Shellbefehl <b>cd</b> nicht ausgeführt werden.
<b>echo</b> <i>Zeichenfolge</i> . . . ]	Schreibt Zeichenfolgen in die Standardausgabe. Informationen zur Verwendung und zu Parametern sind im Abschnitt zum Befehl <b>echo</b> enthalten. Das Flag <b>-n</b> wird nicht unterstützt.
<b>eval</b> [ <i>Argument</i> . . . ]	Liest die Argumente als Eingabe für die Shell und führt die entsprechenden Befehle aus.
<b>exec</b> [ <i>Argument</i> . . . ]	Führt an Stelle der Shell den im Parameter <i>Argument</i> angegebenen Befehl aus, ohne einen neuen Prozess zu erstellen. Eingabe- und Ausgabeargumente können verwendet werden und, sofern keine weiteren Argumente vorhanden sind, eine Änderung der Shelleingabe oder -ausgabe bewirken. Die Verwendung dieses Befehls empfiehlt sich nicht für die Anmeldeshell.
<b>exit</b> [ <i>n</i> ]	Führt zur Beendigung der Shell mit dem im Parameter <i>n</i> angegebenen Rückkehrcode. Wird dieser Parameter nicht angegeben, ist der Rückkehrcode der des zuletzt ausgeführten Befehls (mit der Tastenfolge Strg-D kann die Shell ebenfalls verlassen werden). Der Wert des Parameters <i>n</i> kann zwischen 0 und 255 (einschließlich) liegen.
<b>export</b> [ <i>Name</i> . . . ]	Markiert die angegebenen Namen für den automatischen Export in die Umgebung von nachfolgend ausgeführten Befehlen. Wenn Sie den Parameter <i>Name</i> nicht angeben, zeigt der Befehl <b>export</b> eine Liste aller Namen an, die in diese Shell exportiert wurden. Die Namen von Funktionen können nicht exportiert werden.
<b>hash</b> [ <b>-r</b> ] [ <i>Befehl</i> . . . ]	Sucht und speichert die Position im Suchpfad jedes angegebenen <i>Befehls</i> . Mit dem Flag <b>-r</b> wird die Shell angewiesen, alle gespeicherten Positionen zu verwerfen. Wenn Sie kein Flag und keinen Befehl angeben, zeigt die Shell Informationen über die gespeicherten Befehle im folgenden Format an: <pre>Treffer    Kosten    Befehl</pre> Die Ausführungsanzahl zeigt an, wie oft ein Befehl vom Shellprozess ausgeführt wurde. Kosten ist der berechnete Aufwand für das Auffinden des Befehls im Suchpfad. <i>Befehl</i> gibt den Pfadnamen des angegebenen Befehls an. In bestimmten Fällen ist es erforderlich, dass der Speicherort eines Befehls erneut ermittelt wird, beispielsweise der Standort eines relativen Pfadnamens, wenn das aktuelle Verzeichnis wechselt. Die betreffenden Befehle werden durch einen Stern (*) neben der Information <i>Ausführungsanzahl</i> gekennzeichnet. Der Suchaufwand erhöht sich durch die erneute Ermittlung des Speicherorts.
<b>pwd</b>	Zeigt das aktuelle Verzeichnis an. Eine Beschreibung der Flags für Befehle ist im Abschnitt zum Befehl <b>pwd</b> enthalten.



Eintrag	Beschreibung
<b>read</b> [ <i>Name</i> . . . ]	Liest eine Zeile von der Standardeingabe. Ordnet dem ersten Parameter <i>Name</i> das erste Wort dieser Zeile zu, dem zweiten Parameter <i>Name</i> das zweite Wort usw. Verbleibende Wörter werden dem letzten Parameter <i>Name</i> zugeordnet. Dieser Befehl gibt den Wert 0 zurück, sofern er kein Dateiondezeichen findet.
<b>readonly</b> [ <i>Name</i> . . . ]	Markiert die im Parameter <i>Name</i> angegebenen Namen als schreibgeschützt. Die Werte des Namens können nicht zurückgesetzt werden. Ist kein <i>Name</i> angegeben, wird mit dem Befehl <b>readonly</b> eine Liste aller schreibgeschützten Namen angezeigt.
<b>return</b> [ <i>n</i> ]	Führt zum Verlassen einer Funktion mit dem Rückgabewert <i>n</i> . Wird die Variable <i>n</i> nicht angegeben, gibt die Funktion den Status des letzten innerhalb der Funktion ausgeführten Befehls zurück. Dieser Befehl gilt nur für die Ausführung in einer Shellfunktion.
<b>set</b> [ <i>Flag</i> [ <i>Argument</i> ] . . . ]	<p>Setzt eines oder mehrere der folgenden Flags:</p> <ul style="list-style-type: none"> <li>-a Markiert alle Variablen, für die eine Zuordnung vorgenommen wurde, zum Export. Ist die Zuordnung dem Befehlsnamen vorangestellt, ist das Exportattribut nur für die Ausführungsumgebung dieses Befehls gültig, sofern die Zuordnung keinem der integrierten Sonderbefehle vorangeht. In diesem Fall bleibt das Exportattribut auch nach der Ausführung des integrierten Befehls bestehen. Ist die Zuordnung keinem Befehlsnamen vorangestellt, oder ist die Zuordnung das Verarbeitungsergebnis des Befehls <b>getopts</b> oder <b>read</b>, bleibt das Exportattribut gültig, bis die Variable inaktiviert wird.</li> <li>-e Bricht die Ausführung sofort ab, wenn alle der folgenden Bedingungen für einen Befehl zutreffen: <ul style="list-style-type: none"> <li>• Der Rückgabewert des Befehls ist größer als 0 (null).</li> <li>• Der Befehl ist nicht Teil einer zusammengesetzten Liste der Befehle <b>while</b>, <b>until</b> und <b>if</b>.</li> <li>• Der Befehl wurde nicht mit AND- oder OR-Listen getestet.</li> <li>• Es handelt sich nicht um eine Befehlskette, der das reservierte Wort ! (Ausrufezeichen) vorangestellt ist.</li> </ul> </li> <li>-f Inaktiviert die Dateinamensubstitution.</li> <li>-h Sucht und speichert die Befehle, die innerhalb von Funktionen aufgerufen werden, während diese Funktionen definiert werden. (Normalerweise werden diese Befehle zur Zeit der Funktionsausführung gesucht, siehe Befehl <b>hash</b>.)</li> <li>-k Stellt alle Schlüsselwortparameter in die Umgebung eines Befehls, nicht nur die, die dem Befehlsnamen vorangestellt sind.</li> <li>-n Liest Befehle, führt sie aber nicht aus. Verwenden Sie das Flag <b>-n</b>, um Fehler in der Syntax des Shell-Scripts zu suchen.</li> <li>-t Bricht die Ausführung nach dem Lesen und Ausführen eines Befehls ab.</li> <li>-u Behandelt eine nicht gesetzte Variable als Fehler und bricht die Ausführung der Variablensubstitution sofort ab. Eine interaktive Shell wird nicht verlassen.</li> <li>-v Zeigt die Shellingabezeilen an, wenn sie gelesen werden.</li> <li>-x Zeigt Befehle und die zugehörigen Argumente an, bevor sie ausgeführt werden.</li> <li>— Ändert keines der Flags. Dies ist nützlich, wenn der positionsgebundene Parameter <b>\$1</b> auf eine Zeichenfolge gesetzt wird, die mit einem Minuszeichen (-) beginnt.</li> </ul> <p>Die Verwendung eines Pluszeichens (+) an Stelle eines Minuszeichens (-) hebt die Definition von Flags auf. Diese Flags können auch in der Shellbefehlszeile angegeben werden. Die Sondervariable \$- enthält den aktuellen Satz von Flags.</p> <p>Alle mit dem Befehl <b>set</b> angegebenen <i>Argumente</i> sind positionsgebundene Parameter und werden in der angegebenen Reihenfolge \$1, \$2, . . . usw. zugeordnet. Sind keine <i>Flags</i> oder <i>Argumente</i> angegeben, werden mit dem Befehl <b>set</b> alle Namen und Werte der aktuellen Shellvariablen angezeigt.</p>

Eintrag	Beschreibung
<b>shift</b> [ <i>n</i> ]	<p>Verschiebt Befehlszeilenargumente nach links, d. h. ordnet die Werte der positionsgebundenen Parameter erneut zu, indem der aktuelle Wert von <b>\$1</b> gelöscht wird und <b>\$2</b> der Wert von <b>\$1</b>, <b>\$3</b> der Wert von <b>\$2</b> usw. zugeordnet wird. Sind mehr als 9 Befehlszeilenargumente vorhanden, wird das zehnte Argument <b>\$9</b> zugeordnet. Die verbleibenden Argumente werden vorerst nicht zugeordnet (bis ein weiterer Befehl <b>shift</b> eingegeben wird). Sind neun oder weniger Argumente vorhanden, inaktiviert der Befehl <b>shift</b> den positionsgebundenen Parameter mit der höchsten Nummer, der einen Wert hat.</p> <p>Der positionsgebundene Parameter <b>\$0</b> wird nie verschoben. Der Befehl <b>shift n</b> ist eine verkürzte Schreibweise, mit der die Anzahl <i>n</i> der aufeinanderfolgenden Verschiebungen angegeben wird. Der Standardwert des Parameters <i>n</i> ist 1.</p>
<b>test</b> <i>Ausdruck</i>   [ <i>Ausdruck</i> ]	<p>Wertet Bedingungsausdrücke aus. Eine Beschreibung der Flags und Parameter für Befehle ist im Abschnitt zum Befehl <b>test</b> enthalten. Das Flag <b>-h</b> wird vom integrierten Bourne-Shellbefehl <b>test</b> nicht unterstützt.</p>
<b>times</b>	<p>Zeigt die aufgelaufenen Benutzer- und Systemzeiten für Prozesse an, die von der Shell ausgeführt wurden.</p>
<b>trap</b> [ <i>Befehl</i> ] [ <i>n</i> ] . . .	<p>Führt den im Parameter <i>Befehl</i> angegebenen Befehl aus, wenn die Shell das bzw. die im Parameter <i>n</i> angegebenen Signal(e) empfängt. Die Befehle <b>trap</b> werden in der Reihenfolge der Signalnummern ausgeführt. Jeder Versuch, einen Trap für ein Signal zu setzen, das beim Starten der aktuellen Shell ignoriert wurde, ist wirkungslos.</p> <p><b>Anmerkung:</b> Die Shell liest den Parameter <i>Befehl</i> einmal, wenn der Trap gesetzt wird, und ein weiteres Mal, wenn der Trap ausgeführt wird. Wenn Sie keinen Befehl angeben, werden alle mit dem Parameter <i>n</i> angegebenen Traps auf die aktuellen Werte zurückgesetzt. Wird eine leere Zeichenfolge angegeben, wird dieses Signal von der Shell und den aufgerufenen Befehlen ignoriert. Wenn der Parameter <i>n</i> gleich 0 (null) ist, wird der angegebene Befehl beim Verlassen der Shell ausgeführt. Wenn Sie weder einen Befehl noch ein Signal angeben, wird mit dem Befehl <b>trap</b> eine Liste der Befehle angezeigt, die den einzelnen Signalnummern zugeordnet sind.</p>
<b>type</b> [ <i>Name</i> . . . ]	<p>Zeigt für jeden angegebenen <i>Namen</i> an, wie die Shell ihn als Befehlsnamen interpretieren würde.</p>

Eintrag	Beschreibung
<b>ulimit</b> [-HS] [-c   -d   -f   -m   -r   -s   -t   -u] [Grenzwert]	<p>Zeigt zugeordnete Shellressourcen an oder passt diese an. Die Einstellungen der Shellressource können einzeln oder als Gruppe angezeigt werden. Standardmäßig werden die veränderlichen Ressourceneinstellungen als Gruppe angezeigt.</p> <p>Die Einstellung der Shellressourcen ist abhängig von der in der aktuellen Shell verwendeten Benutzer-ID. Die unveränderlichen Einstellungen einer Ressource können nur von Root festgelegt werden. Beim Versuch, die unveränderlichen Ressourceneinstellungen über eine andere Benutzer-ID festzulegen, wird ein Fehler ausgegeben. Standardmäßig werden von Root sowohl die unveränderlichen als auch die veränderlichen Grenzwerte einer bestimmten Ressource festgelegt. Daher ist bei der Verwendung der Flags <b>-S</b>, <b>-H</b> oder des Standard-Flag für die Festlegung der Grenzwerte durch Root Vorsicht geboten. Andere Benutzer als Root können nur die veränderlichen Grenzwerte einer Ressource festlegen. Wurde für einen Grenzwert von einem anderen Benutzer als Root ein niedrigerer Wert angegeben, kann dieser Wert nicht wieder erhöht werden; auch nicht auf den ursprünglichen Grenzwert des Systems.</p> <p>Zum Festlegen einer Ressourcengrenze wählen Sie das entsprechende Flag und den Grenzwert (eine ganze Zahl) für die neue Ressource aus. Ressourcengrenzen können nur nacheinander definiert werden. Sind mehrere Ressourcen-Flags angegeben, erhalten Sie unbestimmte Ergebnisse. Standardmäßig wird durch Eingabe des Befehls <b>ulimit</b> und nur eines neuen Werts in der Befehlszeile die Dateigröße der Shell festgelegt. Die Angabe des Flags <b>-f</b> ist optional.</p> <p>Folgende Flags können mit dem Befehl <b>ulimit</b> angegeben werden:</p> <ul style="list-style-type: none"> <li><b>-c</b> Das Basiselement für die Shell wird festgelegt oder angezeigt.</li> <li><b>-d</b> Das Datensegment für die Shell wird festgelegt oder angezeigt.</li> <li><b>-f</b> Die Dateigröße für die Shell wird festgelegt oder angezeigt.</li> <li><b>-H</b> Die unveränderliche Ressourcengrenze wird festgelegt oder angezeigt (nur Root).</li> <li><b>-m</b> Der Speicher für die Shell wird festgelegt oder angezeigt.</li> <li><b>-r</b> Die maximale Anzahl Threads pro Prozess wird festgelegt oder angezeigt.</li> <li><b>-s</b> Das Stapelsegment für die Shell wird festgelegt oder angezeigt.</li> <li><b>-S</b> Die veränderliche Ressourcengrenze wird festgelegt oder angezeigt.</li> <li><b>-t</b> Die maximale CPU-Zeit für die Shell wird festgelegt oder angezeigt.</li> <li><b>-u</b> Die maximale Anzahl Prozesse pro Benutzer wird festgelegt oder angezeigt.</li> </ul>
<b>umask</b> [nmn]	Legt die Dateiberechtigungen fest. Beim Erstellen einer Datei werden mit diesem Wert zusammen mit den Berechtigungen des Erstellungsprozesses die Berechtigungen der Datei festgelegt. Der Standardwert ist 022. Ist kein Wert eingegeben, zeigt der Befehl <b>umask</b> den aktuellen Wert an.
<b>unset</b> [Name . . .]	Löscht die entsprechende Variable oder Funktion für jeden im Parameter <i>Name</i> angegebenen Namen. Die Shellvariablen <i>PATH</i> , <i>PS1</i> , <i>PS2</i> , <i>MAILCHECK</i> und <i>IFS</i> können nicht inaktiviert werden.
<b>wait</b> [n]	Wartet auf die Beendigung eines untergeordneten Prozesses, dessen Prozessnummer im Parameter <i>n</i> angegeben ist, und meldet dessen Exit-Status. Wird der Parameter <i>n</i> nicht angegeben, wartet die Shell auf alle momentan aktiven Prozesse und gibt den Rückgabewert null (0) aus.

### Befehlssubstitution in der Bourne-Shell:

Mit der Befehlssubstitution ist es möglich, die Ausgabe jedes Befehls als Argument für einen anderen Befehl zu erfassen.

Wird eine Befehlszeile zwischen umgekehrte Anführungszeichen (` `) gesetzt, führt die Shell zuerst den Befehl bzw. die Befehle aus und ersetzt anschließend den gesamten Ausdruck, einschließlich der umgekehrten Anführungszeichen, durch die Ausgabe. Diese Funktion wird oft in Zuordnungsanweisungen verwendet. So wird beispielsweise bei der Anweisung

```
today=`date`
```

die Zeichenfolge, die das aktuelle Datum darstellt, der Variablen *today* zugeordnet. Mit der folgenden Zuordnung wird die Anzahl der im aktuellen Verzeichnis enthaltenen Dateien in der Variablen *Dateien* gespeichert:

```
Dateien=`ls | wc -l`
```

Die Befehlssubstitution kann für jeden Befehl ausgeführt werden, der in die Standardausgabe schreibt.

Um Befehlssubstitutionen zu verschachteln, muss jedem der verschachtelten umgekehrten Anführungszeichen ein Backslash (\) vorangestellt werden. Beispiel:

```
logmsg=`echo Anmeldeverzeichnis: `pwd``
```

Mit dem Sonderbefehl **read** können den Shellvariablen auch indirekt Werte zugeordnet werden. Dieser Befehl liest eine Zeile der Standardeingabe (gewöhnlich die Tastatur) und ordnet allen benannten Variablen nacheinander Wörter dieser Zeile zu. Beispiel:

```
read Vorname Nachname
```

Dieser Befehl akzeptiert eine Eingabezeile im Format:

```
John Public
```

Das Ergebnis ist dasselbe, als würden Sie Folgendes eingeben:

```
Vorname=John Nachname=Public
```

Der Sonderbefehl **read** ordnet alle überschüssigen Wörter der letzten Variablen zu.

*Variablensubstitution in der Bourne-Shell:*

In der Bourne-Shell können Sie Variablensubstitutionen durchführen.

Die Bourne-Shell verfügt über verschiedene Möglichkeiten, Variablen zu erstellen (einem Namen einen Zeichenfolgewart zuordnen). Bestimmte Variablen, positionsgebundene Parameter und Schlüsselwortparameter werden normalerweise nur in einer Befehlszeile gesetzt. Andere Variablen sind einfache Namen, denen der Benutzer oder die Shell Zeichenfolgewart zuordnen kann.

### **Zugehörige Konzepte:**

„Unbeaufsichtigte Terminals“ auf Seite 312

Alle Systeme bieten Schwachstellen, wenn Terminals mit aktiven Anmeldungen unbeaufsichtigt bleiben. Schwerwiegende Probleme treten auf, wenn der Systemverwalter ein Terminal unbeaufsichtigt lässt, das mit Rootberechtigung aktiviert wurde. Im Allgemeinen sollten sich Benutzer jedes Mal abmelden, wenn sie ihr Terminal verlassen.

### **Benutzerdefinierte Variablen in der Bourne-Shell:**

Die Bourne-Shell erkennt alphanumerische Variablen, denen Zeichenfolgewart zugeordnet werden können.

Geben Sie Folgendes ein, um einem Namen einen Zeichenfolgewart zuzuordnen:

```
Name=Zeichenfolge
```

Ein Name ist eine Folge von Buchstaben, Ziffern und Unterstreichungszeichen, die mit einem Unterstreichungszeichen oder einem Buchstaben beginnt. Um den einer Variablen zugeordneten Wert zu verwenden, müssen Sie am Anfang des Namens ein Dollarzeichen (\$) hinzufügen. Damit erhält die Variable *\$Name* den mit der Variablen *Zeichenfolge* angegebenen Wert. Es ist zu beachten, dass in einer Zuordnungsanweisung keine Leerzeichen rechts und links vom Gleichheitszeichen (=) stehen dürfen. Positionsgebundene Parameter können in einer Zuordnungsanweisung nicht verwendet werden. In einer Befehlszeile können mehrere Zuordnungen angegeben werden, wobei jedoch zu beachten ist, dass die Shell die Zuordnungen von rechts nach links ausführt.

Wird die Variable *Zeichenfolge* in doppelte (") oder einfache (') Anführungszeichen eingeschlossen, interpretiert die Shell Leerzeichen, Tabulatorzeichen, Semikolons und Zeilenvorschubzeichen innerhalb der Zeichenfolge nicht als Wortbegrenzer, sondern fügt sie unverändert in die Zeichenfolge ein.

Wenn Sie die Variable *Zeichenfolge* in doppelte Anführungszeichen (") einschließen, erkennt die Shell die Variablennamen in der Zeichenfolge und führt eine Variablensubstitution aus, d. h., sie ersetzt Verweise auf positionsgebundene Parameter und andere Variablennamen, denen ein Dollarzeichen (\$) vorangestellt ist, durch die entsprechenden Werte, sofern vorhanden. Die Shell führt die Befehlssubstitution auch innerhalb von Zeichenfolgen aus, die in doppelte Anführungszeichen gesetzt sind.

Wenn Sie die Variable *Zeichenfolge* in einfache Anführungszeichen (') einschließen, ersetzt die Shell keine Variablen oder Befehle innerhalb der Zeichenfolge. Das folgende Beispiel verdeutlicht diesen Unterschied:

```
You: num=875
 number1="Add $num"
 number1="Add $num"
 number2='Add $num'
System: Add 875
You: echo $number2
System: Add $num
```

Nach der Variablensubstitution interpretiert die Shell die Leerzeichen in Zuordnungen nicht erneut. Demzufolge ergibt sich bei den folgenden Zuordnungen mit dem Ergebnis `$first` und `$second` derselbe Wert:  
`first='eine Zeichenfolge mit eingebetteten Leerzeichen'`  
`second=$first`

Wird auf eine Variable verwiesen, kann der Variablenname (oder die Ziffer für den positionsgebundenen Parameter) in geschweifte Klammern ({} ) eingeschlossen werden, um den Variablennamen von einer nachfolgenden Zeichenfolge abzugrenzen. Die geschweiften Klammern sind insbesondere dann erforderlich, wenn es sich bei dem unmittelbar auf den Namen folgenden Zeichen um einen Buchstaben, eine Ziffer oder ein Unterstrichungszeichen handelt und die Variable kein positionsgebundener Parameter ist:

```
You: a='This is a'
 echo "${a}n example"
System: This is an example
You: echo "$a test"
System: This is a test
```

### **Zugehörige Konzepte:**

„Positionsgebundene Parameter in der Bourne-Shell“ auf Seite 274

Wird eine Shellprozedur ausgeführt, erstellt die Shell implizit positionsgebundene Parameter, die auf jedes Wort in der Befehlszeile über seine dortige Position verweisen.

### **Zugehörige Verweise:**

„Bedingte Substitution in der Bourne-Shell“ auf Seite 273

Normalerweise ersetzt die Shell den Ausdruck `$Variable` durch den Zeichenfolgewert, der der Variablen *Variable*, falls vorhanden, zugeordnet ist. Es gibt jedoch eine spezielle Schreibweise, mit der eine *bedingte Substitution* ausgeführt werden kann, abhängig davon, ob die Variable gesetzt ist und/oder nicht leer ist.

*Von der Bourne-Shell verwendete Variablen:*

Die Shell verwendet die folgenden Variablen. Während die Shell nur einige dieser Variablen setzt, kann der Benutzer alle Variablen setzen und wieder zurücksetzen.

Eintrag	Beschreibung
<i>CDPATH</i>	Zeigt den Suchpfad für den Befehl <b>cd</b> (change directory, Verzeichnis wechseln) an.
<i>HOME</i>	Zeigt den Namen des <i>Anmeldeverzeichnisses</i> an, das nach Abschluss der Anmeldung zum aktuellen Verzeichnis wird. Diese Variable wird vom Programm <i>login</i> initialisiert. Der Befehl <b>cd</b> verwendet den Wert der Variablen <i>\$HOME</i> als Standardwert. Durch die Verwendung dieser Variablen an Stelle eines expliziten Pfadnamens in einer Shellprozedur kann die Prozedur ohne Änderungen aus verschiedenen Verzeichnissen gestartet werden.
<i>IFS</i>	Die Zeichen, bei denen es sich um interne Feldtrennzeichen handelt, d. h. die Zeichen, die die Shell bei der Interpretation der Leerzeichen verwendet. Die Shell setzt die Variable <i>IFS</i> zunächst, um Leerzeichen, Tabulatorzeichen und Zeilenvorschubzeichen einzuschließen.
<i>LANG</i>	Bestimmt die Locale, die verwendet werden soll, wenn weder in der Variablen <i>LC_ALL</i> noch in der entsprechenden Umgebungsvariablen (die mit <i>LC_</i> beginnt) eine Locale angegeben ist.
<i>LC_ALL</i>	Bestimmt die Locale, mit der alle länderspezifischen Angaben außer Kraft gesetzt werden sollen, die durch die Einstellungen der Umgebungsvariablen <i>LANG</i> oder anderen Umgebungsvariablen, die mit <i>LC_</i> beginnen, festgelegt sind.
<i>LC_COLLATE</i>	Definiert die Sortierfolge, die verwendet werden soll, wenn Namen sortiert werden und Zeichenbereiche in Mustern auftreten.
<i>LC_CTYPE</i>	Bestimmt die Locale für die Interpretation der Bytefolgen von Textdaten als Zeichen (d. h. Einzelbytezeichen gegen Mehrbytezeichen in Argumenten und Eingabedateien). Bestimmt ferner, welche Zeichen als Buchstaben ( <b>Alphazeichenklasse</b> ) definiert sind und wie sich Zeichenklassen bei der Mustererkennung verhalten.
<i>LC_MESSAGES</i>	Bestimmt, in welcher Sprache Nachrichten geschrieben werden.
<i>LIBPATH</i>	Gibt den Suchpfad für gemeinsam benutzte Bibliotheken an.
<i>LOGNAME</i>	Gibt den Anmeldenamen an, der als schreibgeschützt (readonly) in der Datei <i>/etc/profile</i> markiert ist.
<i>MAIL</i>	Zeigt den Pfadnamen der Datei an, die vom elektronischen Mail-System verwendet wird, um neue Mails zu finden. Ist diese Variable gesetzt, überprüft die Shell in regelmäßigen Abständen die Änderungszeit der Datei und zeigt den Wert von <i>\$MAILMSG</i> an, wenn sich die Zeit geändert hat und die Dateilänge größer als 0 ist. Die Variable <i>MAIL</i> wird in der Datei <i>.profile</i> gesetzt. Benutzer des Befehls <b>mail</b> ordnen dieser Variablen normalerweise den Wert <i>/usr/spool/mail/\$LOGNAME</i> zu.
<i>MAILCHECK</i>	Gibt die Anzahl der Sekunden an, die die Shell vergehen lässt, bevor erneut geprüft wird, ob neue Mails in den Dateien eingegangen sind, die in den Variablen <i>MAILPATH</i> oder <i>MAIL</i> angegeben sind. Der Standardwert ist 600 Sekunden (10 Minuten). Wird die Variable <i>MAILCHECK</i> auf 0 gesetzt, führt die Shell vor jeder Eingabeaufforderung eine Überprüfung durch.
<i>MAILMSG</i>	Die Mail-Benachrichtigungsnachricht. Wird die Variable <i>MAILMSG</i> explizit auf eine leere Zeichenfolge ( <i>MAILMSG=""</i> ) gesetzt, wird keine Nachricht angezeigt.
<i>MAILPATH</i>	Eine Liste mit Dateinamen, die durch Doppelpunkte getrennt sind. Ist diese Variable gesetzt, benachrichtigt die Shell den Benutzer, wenn in einer der in der Liste angegebenen Dateien Mails eingegangen sind. Nach jedem Dateinamen kann ein % (Prozentzeichen) und eine Nachricht eingegeben werden, die bei der Ankunft von Mails angezeigt wird. Andernfalls verwendet die Shell den Wert der Variablen <i>MAILMSG</i> oder standardmäßig die Nachricht <i>[SIE HABEN NEUE MAIL]</i> . <b>Anmerkung:</b> Ist die Variable <i>MAILPATH</i> gesetzt, werden die dort angegebenen Dateien an Stelle der in der Variablen <i>MAIL</i> angegebenen Dateien überprüft. Sollen sowohl die Dateien der Variablen <i>MAILPATH</i> als auch die von der Variablen <i>MAIL</i> gesetzte Datei überprüft werden, muss die <i>MAIL</i> -Datei in der Liste der <i>MAILPATH</i> -Dateien angegeben sein.

Eintrag	Beschreibung
<i>PATH</i>	<p>Der Suchpfad für Befehle. Es handelt sich dabei um eine geordnete Liste mit Verzeichnispfadnamen, die durch Doppelpunkte getrennt sind. Die Shell durchsucht diese Verzeichnisse in der angegebenen Reihenfolge, wenn sie nach Befehlen sucht. Das aktuelle Verzeichnis wird in der Liste durch eine leere Zeichenfolge dargestellt.</p> <p>Die Variable <i>PATH</i> wird normalerweise in der Datei <i>/etc/environment</i> initialisiert, und zwar mit <i>/usr/bin:/etc:/usr/sbin:/usr/ucb:/usr/bin/X11:/sbin</i>. Diese Variable kann den eigenen Anforderungen entsprechend zurückgesetzt werden. Die Variable <i>PATH</i>, die sich in der Datei <i>.profile</i> befindet, enthält auch das Verzeichnis <i>\$HOME/bin</i> und das aktuelle Verzeichnis.</p> <p>Ist ein projektspezifisches Befehlsverzeichnis vorhanden, z. B. <i>/project/bin</i>, das vor den Standardsystemverzeichnissen durchsucht werden soll, müssen Sie die Variable <i>PATH</i> wie folgt angeben:  <i>PATH=/project/bin:\$PATH</i></p> <p>Soll die Variable <i>PATH</i> auf einen anderen als den Standardwert gesetzt werden, wird dies am besten in der Datei <i>\$HOME/.profile</i> ausgeführt. Die Variable <i>PATH</i> kann nicht zurückgesetzt werden, wenn Befehle unter der Restricted Shell ausgeführt werden.</p>
<i>PS1</i>	Die als Primäreingabeaufforderung zu verwendende Zeichenfolge. Eine interaktive Shell zeigt diese Eingabeaufforderung an, wenn eine Eingabe erwartet wird. Der Standardwert der Variablen <i>PS1</i> (für andere Benutzer als Root) ist ein Dollarzeichen (\$) mit nachfolgendem Leerzeichen.
<i>PS2</i>	Der Wert der Sekundäreingabeaufforderung. Erwartet die Shell weitere Eingaben, wenn sie auf ein Zeilenvorschubzeichen trifft, wird als Eingabeaufforderung der Wert der Variablen <i>PS2</i> angezeigt. Der Standardwert der Variablen <i>PS2</i> ist > mit nachfolgendem Leerzeichen.
<i>SHACCT</i>	Gibt den Namen einer Datei an, deren Eigner der Benutzer ist. Ist die Variable gesetzt, schreibt die Shell einen Abrechnungssatz für jedes ausgeführte Shell-Script in die Datei. Um die gesammelten Daten zu analysieren, können Abrechnungsprogramme, wie z. B. <i>acctcom</i> und <i>acctcms</i> , verwendet werden.
<i>SHELL</i>	Der Pfadname der Shell, die in der Umgebung existiert. Diese Variable sollte von der Datei <i>\$HOME/.profile</i> jeder eingeschränkten Anmeldung gesetzt und exportiert werden.
<i>TIMEOUT</i>	Die Anzahl an Minuten, die die Shell vor dem Verlassen inaktiv bleibt. Ist diese Variable auf einen Wert größer als 0 (null) gesetzt, wird die Shell verlassen, wenn innerhalb der vorgeschriebenen Anzahl von Sekunden nach Ausgabe der Eingabeaufforderung <i>PS1</i> kein Befehl eingegeben wurde. (Es ist zu beachten, dass die Shell mit einem maximalen Grenzwert kompiliert werden kann, der für den hier beschriebenen Wert nicht überschritten werden darf.) Der Wert null gibt an, dass kein Zeitlimit vorhanden ist.

### Zugehörige Konzepte:

„Interpretation von Leerzeichen“ auf Seite 288

Nach der Variablen- und Befehlssubstitution durchsucht die Shell die Ergebnisse nach internen Feldtrennzeichen (d. h. nach den Zeichen, die in der Variable *IFS* definiert sind).

*Vordefinierte Sondervariablen in der Bourne-Shell:*

Verschiedene Variablen haben eine Sonderbedeutung. Die folgenden Variablen werden nur von der Bourne-Shell gesetzt:

Eintrag	Beschreibung
<i>\$@</i>	<p>Erweitert die positionsgebundenen Parameter, beginnend mit <b>\$1</b>. Jeder Parameter wird durch ein Leerzeichen abgetrennt.</p> <p>Wenn Sie <b>\$@</b> in doppelte Anführungszeichen (" ") setzen, interpretiert die Shell jeden positionsgebundenen Parameter als separate Zeichenfolge. Sind keine positionsgebundenen Parameter vorhanden, erweitert die Shell die Anweisung auf eine nicht in Anführungszeichen gesetzte, leere Zeichenfolge.</p>
<i>\$*</i>	<p>Erweitert die positionsgebundenen Parameter, beginnend mit <b>\$1</b>. Die Shell trennt die Parameter durch das erste Zeichen des Wertes der Variablen <i>IFS</i>.</p> <p>Wenn Sie <b>\$*</b> in doppelte Anführungszeichen (" ") setzen, übernimmt die Shell die Werte der positionsgebundenen Parameter in die Anführungszeichen. Die Werte sind durch das erste Zeichen des Parameters <i>IFS</i> getrennt.</p>

Eintrag	Beschreibung
\$#	Gibt die Anzahl der an die Shell übergebenen positionsgebundenen Parameter an, wobei der Name der Shellprozedur selbst nicht berücksichtigt wird. Die Variable \$# liefert demzufolge die Nummer des positionsgebundenen Parameters mit der höchsten Nummer, der gesetzt wurde. Diese Variable dient hauptsächlich dazu, zu überprüfen, ob die erforderliche Anzahl an Argumenten vorhanden ist. Über die Shell kann nur auf die positionsgebundenen Parameter \$0 bis \$9 zugegriffen werden.
\$?	Gibt den Exit-Wert des zuletzt ausgeführten Befehls an. Der Wert ist eine Dezimalzeichenfolge. Die meisten Befehle geben den Wert 0 zurück, um die erfolgreiche Beendigung anzuzeigen. Die Shell selbst gibt den aktuellen Wert des Parameters \$? als Exit-Wert zurück.
\$\$	Gibt die Prozessnummer des aktuellen Prozesses an. Da die Prozessnummern der vorhandenen Prozesse eindeutig sind, wird diese Zeichenfolge oft für die Generierung eindeutiger Namen für temporäre Dateien verwendet.  Das folgende Beispiel verdeutlicht die empfohlene Vorgehensweise bei der Erstellung temporärer Dateien in einem eigens dafür angelegten Verzeichnis: <pre>temp=/tmp/\$\$ ls &gt;\$temp . . . rm \$temp</pre>
#!	Gibt die Prozessnummer des letzten Prozesses an, der mit dem Abschlusszeichen & im Hintergrund ausgeführt wurde.
\$-	Eine Zeichenfolge, die sich aus den Namen der momentan in der Shell gesetzten Ausführungs-Flags zusammensetzt.

### Zugehörige Konzepte:

„Positionsgebundene Parameter in der Bourne-Shell“ auf Seite 274

Wird eine Shellprozedur ausgeführt, erstellt die Shell implizit positionsgebundene Parameter, die auf jedes Wort in der Befehlszeile über seine dortige Position verweisen.

### Interpretation von Leerzeichen:

Nach der Variablen- und Befehlssubstitution durchsucht die Shell die Ergebnisse nach internen Feldtrennzeichen (d. h. nach den Zeichen, die in der Variable *IFS* definiert sind).

Die Shell teilt die Zeile an den Stellen, an denen solche Zeichen auftreten, in einzelne Wörter ein, wobei die einzelnen Wörter durch ein Leerzeichen voneinander getrennt werden. Explizite Nullargumente (" oder ') werden beibehalten, und implizite Nullargumente (die sich aus Parametern ohne Werte ergeben) werden verworfen.

### Zugehörige Verweise:

„Von der Bourne-Shell verwendete Variablen“ auf Seite 285

Die Shell verwendet die folgenden Variablen. Während die Shell nur einige dieser Variablen setzt, kann der Benutzer alle Variablen setzen und wieder zurücksetzen.

## C-Shell

Die C-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlsorientierte Programmiersprache. Die verwendete Syntax ist der Programmiersprache C ähnlich.

Der Befehl **csh** startet die C-Shell.

Wenn Sie sich anmelden, durchsucht der Befehl **csh** zuerst die systemweite Konfigurationsdatei `/etc/csh.cshrc`. Ist diese Konfigurationsdatei vorhanden, führt die C-Shell die in dieser Datei gespeicherten Befehle aus. Anschließend führt die C-Shell die systemweite Setup-Datei `/etc/csh.login` aus, falls diese verfügbar ist. Danach werden die Dateien `.cshrc` und `.login` im Ausgangsverzeichnis (home) gesucht. Sind diese Dateien vorhanden, enthalten sie alle angepassten Benutzerdaten, die zur Ausführung der C-Shell gehören. Alle Variablen, die in den Dateien `/etc/csh.cshrc` und `/etc/csh.login` festgelegt sind,



können von den Dateien `.cshrc` und `.login`, die sich im Verzeichnis `$HOME` befinden, überschrieben werden. Nur Root kann die Dateien `/etc/csh.cshrc` und `/etc/csh.login` ändern.

Die Dateien `/etc/csh.login` und `$HOME/.login` werden nur einmal beim Anmelden ausgeführt. Sie werden normalerweise verwendet, um Definitionen für Umgebungsvariablen, Befehle, die beim Anmelden einmalig ausgeführt werden, oder Befehle, mit denen Terminalmerkmale festgelegt werden, zu speichern.

Die Dateien `/etc/csh.cshrc` und `$HOME/.cshrc` werden bei der Anmeldung und bei jedem Aufruf des Befehls `csh` oder eines C-Shell-Script ausgeführt. Sie werden normalerweise verwendet, um C-Shellmerkmale wie Aliasbefehle und C-Shellvariablen (zum Beispiel `history`, `noclobber` oder `ignoreeof`) zu definieren. Es wird empfohlen, nur die integrierten C-Shellbefehle aus den Dateien `/etc/csh.cshrc` und `$HOME/.cshrc` zu verwenden, weil die Verwendung anderer Dateien die Startzeit der Shell-Scripts verlängert.

### Zugehörige Verweise:

„Liste der integrierten C-Shellbefehle“ auf Seite 297

Im Folgenden sind die integrierten C-Shellbefehle aufgelistet.

### Einschränkungen der C-Shell:

Für die C-Shell gelten folgende Einschränkungen.

- Wörter dürfen nicht länger als 1024 Byte sein.
- Argumentlisten sind auf den Wert (Bytes) von `ARG_MAX` beschränkt. Die Werte für die Variable `ARG_MAX` sind in der Datei `/usr/include/sys/limits.h` enthalten.
- Die Anzahl der Argumente für einen Befehl, der sich auf Dateinamenerweiterung bezieht, ist auf 1/6 der in einer Argumentliste erlaubten Bytes begrenzt.
- Die Befehlssubstitutionen können nicht mehr Bytes ersetzen, als in einer Argumentliste erlaubt sind.
- Um Schleifen zu erkennen, begrenzt die Shell die Anzahl der Aliassubstitutionen in einer einzelnen Zeile auf 20.
- Der Befehl `csh` unterstützt keine auf Äquivalenzklassifikation von Zeichen basierende Dateinamenerweiterung.
- Dateideskriptoren (abgesehen von Standardeingabe, Standardausgabe und Standardfehler), die bereits geöffnet waren, bevor `csh` eine Anwendung ausgeführt hat, stehen dieser Anwendung nicht zur Verfügung.

### Aliassubstitution in der C-Shell:

Ein *Aliasname* ist ein Name, der einem Befehl oder einer Befehlsfolge zugeordnet ist. In der C-Shell können Aliasnamen zugeordnet und wie Befehle verwendet werden. Die Shell verwaltet eine Liste der definierten Aliasnamen.

Nachdem die Shell die Befehlszeile durchsucht hat, teilt sie diese in einzelne Wörter auf und prüft das erste Wort jedes Befehls von links nach rechts daraufhin, ob der Befehl einen Aliasnamen enthält. Wird ein Aliasname gefunden, verwendet die Shell den Protokollmechanismus, um den Text des Aliasnamens durch den Text des Befehls zu ersetzen, auf den der Aliasname verweist. Die daraus resultierenden Wörter ersetzen den Befehl und die Argumentenliste. Wird nicht auf die Protokollliste verwiesen, bleibt die Argumentenliste unverändert.

Mit den integrierten Befehlen `alias` und `unalias` können Aliaslisten erstellt, angezeigt und geändert werden. Folgendes Format für den Befehl `alias` verwenden:

```
alias [Name [Wortliste]]
```

Die optionale Variable *Name* gibt den Aliasnamen für den angegebenen Namen an. Wenn Sie mit der Variablen *Wortliste* eine Wortliste angeben, ordnet der Befehl diese Liste der Variablen *Name* als Aliasname zu. Wenn Sie den Befehl `alias` ohne die optionalen Variablen ausführen, zeigt er alle Aliasnamen für die C-Shell an.

Falls für den Befehl `ls` der Aliasname `ls -l` festgelegt ist, wird der Befehl

```
ls /usr
```

durch folgenden Befehl ersetzt:

```
ls -l /usr
```

Die Argumentenliste bleibt unverändert, da im Befehl kein Verweis auf die Protokollliste über einen Aliasnamen vorhanden ist. Ähnliches gilt im folgenden Fall: Ist für den Befehl `lookup` der Aliasname

```
grep \!^ /etc/passwd
```

festgelegt, ersetzt die Shell `lookup bill` durch

```
grep bill /etc/passwd
```

In diesem Beispiel bezieht sich `!^` auf die Protokollliste, und die Shell ersetzt diese Angabe durch das erste Argument in der Eingabezeile, hier `bill`.

In Aliasnamen können Platzhalterzeichen verwendet werden. Der Befehl

```
alias lprint 'pr &slash2.!* >
```

```
> print'
```

erstellt einen Befehl, der seine Argumente auf dem Zeilendrucker ausgibt. Das Zeichen `!` ist innerhalb des Aliasnamens durch einfache Anführungszeichen vor der Shell geschützt, so dass es erst dekomprimiert wird, wenn der Befehl `pr` ausgeführt wird.

Wenn die Shell einen Aliasnamen findet, setzt sie die Wörter des Eingabetextes um und beginnt in der umgestalteten Eingabezeile erneut mit der Verarbeitung der Aliasnamen. Falls das erste Wort des neuen Textes mit dem alten übereinstimmt, können Sie eine Schleife dadurch verhindern, dass Sie den Aliasnamen kennzeichnen, damit der Aliasprozess beendet wird. Auftretende Schleifen werden erkannt und führen zu einem Fehler.

### Zugehörige Konzepte:

„Protokollsubstitution in der C-Shell“ auf Seite 307

Mit der Protokollsubstitution können einzelne Wörter von vorherigen Befehlen geändert werden, um somit neue Befehle zu erstellen. Die Protokollsubstitution erleichtert die Wiederholung von Befehlen, die Wiederholung von Argumenten eines vorherigen Befehls im aktuellen Befehl oder die Berichtigung von Schreibfehlern im vorherigen Befehl mit geringem Eingabeaufwand.

### Variablensubstitution in der C-Shell:

Die C-Shell verwaltet einen Satz von Variablen, von denen jede über eine Liste von null oder mehr Wörtern als Wert verfügt. Einige dieser Variablen werden von der Shell gesetzt, oder es wird auf sie verwiesen. Die Variable `argv` beispielsweise ist ein Abbild der Shellvariablenliste. Auf Wörter, die den Wert dieser Variablen enthalten, wird auf besondere Weise verwiesen.

Sie können die Werte dieser Variablen mit den Befehlen `set` und `unset` ändern und anzeigen. Bei einigen Variablen, auf die die Shell verweist, handelt es sich um Schalter (Variablen, die eine Funktion ein- oder ausschalten). Die Shell untersucht keine Schalter für einen Wert, sondern nur, ob sie definiert sind oder nicht. Die Shellvariable `verbose` z. B. ist ein Schalter, der dafür sorgt, dass die Befehlseingabe zurückgemeldet wird. Diese Variable wird durch Eingabe des Flags `-v` in der Befehlszeile gesetzt.

Andere Operationen verarbeiten Variablen numerisch. Der Befehl `@` führt numerische Berechnungen aus, deren Ergebnis einer Variablen zugeordnet wird. Variablenwerte werden jedoch immer als (leere oder nicht leere) Zeichenfolgen dargestellt. In numerischen Operationen ist eine leere Zeichenfolge gleichbedeutend mit null. Bei Werten mit mehreren Wörtern werden das zweite sowie alle nachfolgenden Wörter ignoriert.

Nach der Eingabe eines Befehls analysiert die Shell die Eingabezeile syntaktisch und führt die Aliassubstitution durch. Anschließend führt sie die Variablensubstitution durch, bevor der Befehl ausgeführt wird. Das Dollarzeichen (\$) ist das Schlüsselwort für die Substitution. Es wird jedoch unverändert übergeben, wenn ihm ein Leerzeichen, Tabulatorzeichen oder Zeilenvorschubzeichen folgt. Durch Voranstellen eines Backslash (\) vor das Dollarzeichen (\$) wird diese Erweiterung verhindert, mit Ausnahme der folgenden beiden Fälle:

- Der Befehl wird in doppelte Anführungszeichen (" ") eingeschlossen. In diesem Fall führt die Shell die Substitution immer aus.
- Der Befehl wird in einfache Anführungszeichen (' ') eingeschlossen. In diesem Fall führt die Shell die Substitution nie aus. Zeichenfolgen, die in einfache umgekehrte Anführungszeichen (` `) eingeschlossen sind, werden für die Befehlssubstitution interpretiert.

Die Shell erkennt die Umleitung von Eingabe und Ausgabe vor der Variablenerweiterung und erweitert sie unabhängig voneinander. Andernfalls werden der Befehlsname und die vollständige Argumentenliste zusammen erweitert. Daher ist es möglich, dass das erste (Befehls-)Wort mehr als ein Wort generiert, von denen das erste zum Befehlsnamen und die restlichen zu Parametern werden.

Sofern keine doppelten Anführungszeichen (" ") gesetzt werden oder der Änderungswert :q nicht angegeben wird, ist es möglich, dass für die Ergebnisse der Variablensubstitution eine Befehls- und Dateinamensubstitution durchgeführt wird. Ist eine Variable in doppelte Anführungszeichen eingeschlossen und besteht ihr Wert aus mehreren Wörtern, wird sie zu einem einzigen Wort oder einem Teil eines Wortes erweitert, wobei die Wörter des Variablenwerts durch Leerzeichen voneinander getrennt sind. Wird der Änderungswert :q auf eine Substitution angewendet, wird die Variable zu mehreren Wörtern erweitert. Jedes Wort wird durch ein Leerzeichen abgetrennt und in doppelte Anführungszeichen gesetzt, um eine spätere Befehls- oder Dateinamensubstitution zu verhindern.

Mit den folgenden Schreibweisen können Variablenwerte in die Shellvorgabe eingeführt werden. Nur wenn es nicht anders angegeben ist, ist es ein Fehler, auf eine Variable zu verweisen, die nicht mit dem Befehl set gesetzt wurde.

Sie können die Änderungswerte :gh, :gt, :gr, :h, :r, :q und :x auf die folgenden Substitutionen anwenden. Werden im Befehlsformat geschweifte Klammern ({} ) verwendet, müssen die Änderungswerte in den geschweiften Klammern stehen. In einer Variablenerweiterung ist jeweils nur ein Doppelpunktänderungswert (:Wert) zulässig.

Eintrag	Beschreibung
\$Name	
\${Name}	Ersetzt durch die Wörter, die der Variablen Name zugeordnet sind, wobei die Wörter durch Leerzeichen voneinander getrennt sind. Durch geschweifte Klammern wird die Variable Name von allen nachfolgenden Zeichen abgetrennt, die andernfalls Teil der Variable wären. Die Namen von Shellvariablen beginnen mit einem Buchstaben und bestehen aus bis zu 20 Buchstaben und Ziffern einschließlich des Unterstrichzeichens (_). Gibt die Variable "Name" keine Shellvariable an, wurde sie aber in der Umgebung gesetzt, wird ihr Wert zurückgegeben. Die Änderungswerte, denen ein Doppelpunkt vorangestellt ist, sowie andere hier beschriebene Formate sind in diesem Fall nicht verfügbar.
\$Name[Nummer]	
\${Name[Nummer]}	Wählt nur einige Wörter aus dem Wert der Variablen Name aus. Die Nummer unterliegt der Variablensubstitution und kann aus einer einzelnen Ziffer oder aus zwei Ziffern bestehen, die durch ein Minuszeichen (-) voneinander getrennt sind. Das erste Wort des Zeichenfolgewerts einer Variablen ist mit 1 nummeriert. Wird die erste Nummer eines Bereichs nicht angegeben, wird standardmäßig 1 verwendet. Wird die letzte Nummer des Bereichs nicht angegeben, wird standardmäßig \$#Name verwendet. Mit dem Symbol (*) (Stern) werden alle Wörter ausgewählt. Ist der Bereich leer, tritt kein Fehler auf, wenn das zweite Argument nicht angegeben wurde oder sich nicht in einem Bereich befindet.
\$#Name	
\${#Name}	Gibt die Anzahl der Wörter in der Variablen Name an. Diese Angabe kann wie oben dargestellt in einer [Nummer] verwendet werden. Beispiel: \$Name[ \$#Name].

Eintrag	Beschreibung
\$0	Ersetzt den Namen der Datei, aus der die Befehlseingabe gelesen wird. Wenn der Name nicht bekannt ist, tritt ein Fehler auf.
$\$Nummer$	
$\${Nummer}$	Entspricht $\$argv[Nummer]$ .
$\$*$	Entspricht $\$argv[*]$ .

Die folgenden Substitutionen können möglicherweise nicht durch Änderungswerte mit einem Doppelpunkt (:) geändert werden:

Eintrag	Beschreibung
$\$?Name$	
$\${?Name}$	Ersetzt die Zeichenfolge 1, wenn die Variable <i>Name</i> gesetzt ist. Ersetzt die Zeichenfolge 0 (null), wenn die Variable nicht gesetzt ist.
$\$?0$	Ersetzt 1, wenn der Name der aktuellen Eingabedatei bekannt ist. Ersetzt 0 (null), wenn der Dateiname nicht bekannt ist.
$\$\$$	Ersetzt die Prozessnummer (dezimal) der übergeordneten Shell.
$\$<$	Ersetzt eine Zeile der Standardeingabe ohne weitere Interpretation. Diese Substitution ist zu verwenden, um in einer Shellprozedur die Eingabe von einer Tastatur zu lesen.

### Zugehörige Konzepte:

„Befehlssubstitution in der C-Shell“ auf Seite 306

Bei der *Befehlssubstitution* führt die Shell den angegebenen Befehl aus und ersetzt den Befehl durch seine Ausgabe.

### Dateinamensubstitution in der C-Shell:

In der C-Shell können Dateinamensubstitutionen durchgeführt werden.

Die C-Shell bietet verschiedene Direktaufrufe, die Ihnen Zeit und Tastenanschläge ersparen. Wenn ein Wort das Zeichen \*, ?, [ ] oder { } enthält oder mit einer Tilde (~) beginnt, ist dieses Wort ein Kandidat für die Dateinamensubstitution. Die C-Shell interpretiert das Wort als Muster und ersetzt es durch eine alphabetisch sortierte Liste mit Dateinamen, die dem Muster entsprechen.

Dazu wird die aktuelle Sortierfolge verwendet, die in der Umgebungsvariablen *LC\_COLLATE* oder *LANG* angegeben ist. Bei einer Liste von Wörtern, die die Dateinamensubstitution angeben, tritt ein Fehler auf, wenn keines der Muster mit einem der vorhandenen Dateinamen übereinstimmt. Es ist jedoch nicht erforderlich, dass für alle Muster eine Übereinstimmung gefunden wird. Nur die Platzhalterzeichen \*, ? und [ ] weisen auf Mustererkennung oder Dateinamenerweiterung hin. Die Tilde (~) und die geschweiften Klammern { } zeigen eine Dateinamenabkürzung an.

### Dateinamenserweiterung in der C-Shell:

Der Stern (\*) entspricht jeder Zeichenfolge, einschließlich der leeren Zeichenfolge.

Beispielsweise werden in einem Verzeichnis mit den Dateien

```
a aa aax alice b bb c cc
```

mit dem Befehl **echo a\*** alle Dateinamen ausgegeben, die mit a beginnen:

```
a aa aax alice
```

**Anmerkung:** Beim Abgleich von Dateinamen müssen die Zeichen Punkt (.) und Schrägstrich (/) explizit übereinstimmen.

Das Fragezeichen (?) ersetzt jedes Zeichen. Der Befehl

```
ls a?x
```

listet alle Dateinamen auf, die mit dem Buchstaben a beginnen, dem ein Einzelzeichen folgt, und die mit dem Buchstaben x enden:

```
aax
```

Um ein Einzelzeichen oder einen Bereich von Zeichen abzugleichen, muss das bzw. müssen die Zeichen in eckige Klammern ([ ]) gesetzt werden. Der Befehl

```
ls [abc]
```

listet alle Dateinamen auf, die mit einem der eingeschlossenen Zeichen genau übereinstimmen:

```
a b c
```

Innerhalb von eckigen Klammern wird ein lexikalischer Bereich von Zeichen durch [a-z] angezeigt. Die Zeichen, die mit diesem Muster übereinstimmen, sind durch die aktuelle Sortierfolge definiert.

### Abkürzung von Dateinamen in der C-Shell:

Die Tilde (~) und die geschweiften Klammern { } zeigen einen abgekürzten Dateinamen an. Eine Tilde (~) am Anfang eines Dateinamens wird zur Darstellung des Ausgangsverzeichnisses verwendet. Steht die Tilde (~) allein, wird sie durch das Ausgangsverzeichnis ersetzt, das über den Wert der Shellvariablen *home* definiert ist.

Der Befehl

```
ls ~
```

beispielsweise listet alle Dateien und Verzeichnisse im Verzeichnis \$HOME auf.

Wird hinter dem Befehl ein Name angegeben, der aus Buchstaben, Ziffern und dem Zeichen - besteht, sucht die Shell nach einem Benutzer mit diesem Namen und ersetzt das Verzeichnis \$HOME dieses Benutzers.

**Anmerkung:** Anmerkung: Folgt auf die Tilde (~) ein anderes Zeichen als ein Buchstabe oder ein Schrägstrich (/), oder kommt eine Tilde an einer anderen Stelle als am Wortanfang vor, findet keine Erweiterung statt.

Um Zeichen in Dateinamen abzugleichen, ohne den gesamten Dateinamen einzugeben, schließen Sie den Dateinamen in geschweifte Klammern ({ }) ein. Das Muster a{b,c,d}e ist eine andere Schreibweise für abe ace ade. Die Shell behält die Reihenfolge von links nach rechts bei und speichert deswegen die Ergebnisse der Abgleiche separat auf einer niedrigen Stufe, um diese Reihenfolge zu erhalten. Dieses Konstrukt kann verschachtelt werden. Die Angabe

```
~source/s1/{oldls,ls}.c
```

wird zu

```
/usr/source/s1/oldls.c /usr/source/s1/ls.c
```

erweitert, wenn das Ausgangsverzeichnis für **source** das Verzeichnis /usr/source ist. Ähnlich wird

```
../{memo,*box}
```

zu Folgendem erweitert:

```
../memo ../box ../mbox
```

**Anmerkung:** memo wird nicht zusammen mit den Ergebnissen des Abgleichs mit \*box sortiert. In diesem Sonderfall werden die Zeichen {, } und { } unverändert übergeben.

## Zeichenklassen in der C-Shell:

Sie können Zeichenklassen verwenden, um Dateinamen in einem angegebenen Bereich abzugleichen.

Mit diesem Format wird das System angewiesen, jedes einzelne Zeichen, das zur angegebenen Klasse gehört, abzugleichen.

[*:Zeichenklasse:*]

Die folgenden Klassen entsprechen den **ctype**-Subroutinen:

Zeichenklasse	Definition
<b>alnum</b>	Alphanumerische Zeichen
<b>alpha</b>	Groß- und Kleinbuchstaben
<b>cntrl</b>	Steuerzeichen
<b>digit</b>	Ziffern
<b>graph</b>	Grafikzeichen
<b>lower</b>	Kleinbuchstaben
<b>print</b>	Druckbare Zeichen
<b>punct</b>	Interpunktionszeichen
<b>space</b>	Leerzeichen, Horizontaltabulatorzeichen, Wagenrücklaufzeichen, Zeilenvorschubzeichen, Vertikaltabulatorzeichen oder Formularvorschubzeichen
<b>upper</b>	Großbuchstaben
<b>xdigit</b>	Hexadezimalzeichen

Beispiel: Der Benutzer befindet sich in einem Verzeichnis mit den folgenden Dateien:

```
a aa aax Alice b bb c cc
```

Geben Sie an der Eingabeaufforderung der C-Shell den folgenden Befehl ein:

```
ls [:lower:]
```

Die C-Shell listet alle Dateinamen auf, die mit einem Kleinbuchstaben beginnen:

```
a aa aax b bb c cc
```

Nähere Informationen zu Ausdrücken mit Zeichenklassen finden Sie in der Beschreibung des Befehls **ed**.

## Umgebungsvariablen in der C-Shell:

Bestimmte Variablen haben in der C-Shell eine Sonderbedeutung. Die Variablen *argv*, *cwd*, *home*, *path*, *prompt*, *shell* und *status* werden immer von der Shell gesetzt.

Mit Ausnahme der Variablen *cwd* und *status* erfolgt die Einstellung durch die Shell nur bei der Initialisierung. Die Einstellungen der genannten Variablen bleiben so lange gültig, bis sie explizit zurückgesetzt werden.

Der Befehl **cs**h kopiert die Umgebungsvariablen *USER*, *TERM*, *HOME* und *PATH* in die *cs*h-Variablen *user*, *term*, *home* bzw. *path*. Die Werte werden jedes Mal, wenn die normalen Shellvariablen zurückgesetzt werden, in die Umgebung zurückkopiert. Die Variable *path* kann nur in der Datei **.cshrc** definiert werden, weil die **cs**h-Subprozesse die Pfadinformationen aus der Umgebung importieren und bei Änderung erneut exportieren.

Die folgenden Variablen haben eine Sonderbedeutung:

<b>Eintrag</b>	<b>Beschreibung</b>
<i>argv</i>	Enthält die Argumente, die an die Shell-Scripts übergeben wurden. Positionsparameter werden von dieser Variablen ersetzt.
<i>cdpath</i>	Gibt eine Liste mit alternativen Verzeichnissen an, die vom Befehl <b>chdir</b> oder <b>cd</b> durchsucht werden sollen, um Unterverzeichnisse zu finden.
<i>cwd</i>	Gibt den vollständigen Pfadnamen des aktuellen Verzeichnisses an.
<i>echo</i>	Wird gesetzt, wenn das Befehlszeilen-Flag <b>-x</b> verwendet wird. Ist die Variable gesetzt, geben alle Befehle und die zugehörigen Argumente kurz vor der Ausführung eine Meldung zurück. Bei nicht integrierten Befehlen werden alle Erweiterungen vor der Rückmeldung ausgeführt. Die Rückmeldung integrierter Befehle erfolgt vor der Befehls- und Dateinamenssubstitution, da diese Substitutionen dann selektiv ausgeführt werden.
<i>histchars</i>	Gibt einen Zeichenfolgewert an, mit dem die in der Protokollsubstitution verwendeten Zeichen geändert werden sollen. Das erste Zeichen dieses Werts wird als Protokollsubstitutionszeichen verwendet und ersetzt das Standardzeichen <b>!</b> . Das zweite Zeichen des Werts ersetzt das Winkelzeichen ( <b>^</b> ) in schnellen Substitutionen. <b>Anmerkung:</b> Wird als Wert für <b>histchars</b> ein Zeichen angegeben, dass in Befehls- oder Dateinamen verwendet wird, kann dies zu einer unbeabsichtigten Protokollsubstitution führen.
<i>history</i>	Enthält einen numerischen Wert, der die Größe der Protokollliste steuert. Alle Befehle, deren Nummer im zulässigen Bereich liegen, werden nicht gelöscht. Wenn Sie einen zu großen Wert für die Variable <i>history</i> festlegen, verfügt die Shell möglicherweise nicht über genügend Speicher. Unabhängig davon, ob diese Variable gesetzt ist, wird der letzte Befehl, der über die Protokollliste ausgeführt wurde, von der C-Shell gespeichert.
<i>home</i>	Zeigt das Ausgangsverzeichnis an, das von der Umgebung initialisiert wurde. Die Dateinamenerweiterung des Tildezeichens ( <b>~</b> ) bezieht sich auf diese Variable.
<i>ignoreeof</i>	Gibt an, dass die Shell ein Dateiendezeichen ignoriert, das über Eingabeeinheiten eingegeben wurde, bei denen es sich um Workstations handelt. Damit wird verhindert, dass Shells versehentlich gelöscht werden, wenn sie ein Dateiendezeichen (Strg-D) lesen.
<i>mail</i>	Gibt die Dateien an, in denen die Shell nach Mails sucht. Dieser Vorgang erfolgt nach jedem Befehlsabschluss und führt zu einer Eingabeaufforderung, wenn ein angegebenes Zeitintervall abgelaufen ist. Die Shell zeigt die Nachricht <b>Mail in file an</b> , wenn die Datei vorhanden ist und die Zeit des letzten Zugriffs vor dem Zeitpunkt der letzten Änderung liegt. Ist das erste Wort des Werts der Variablen <i>mail</i> numerisch, gibt es ein anderes Zeitintervall (in Sekunden) für die Suche nach Mails an. Der Standardwert ist 600 (10 Minuten). Werden mehrere Nachrichtendateien angegeben, zeigt die Shell die Nachricht <b>New mail in file an</b> , wenn sich Mails in der angegebenen Datei befinden.
<i>noclobber</i>	Legt die Einschränkungen für die Ausgabeumleitung fest, um sicherzustellen, dass Dateien nicht versehentlich zerstört werden und dass Umleitungen an vorhandene Dateien angefügt werden.
<i>noglob</i>	Verhindert die Dateinamenerweiterung. Die Verwendung dieser Variablen empfiehlt sich in Shell-Skripten, die sich nicht mit Dateinamen befassen, oder wenn eine Liste mit Dateinamen erhalten wurde und keine zusätzlichen Erweiterungen erwünscht sind.
<i>nomatch</i>	Gibt an, dass kein Fehler auftritt, wenn eine Dateinamenerweiterung mit keiner der vorhandenen Dateien übereinstimmt. Stattdessen wird das Elementarmuster zurückgegeben. Es bleibt weiterhin ein Fehler, wenn das Elementarmuster fehlerhaft ist.
<i>notify</i>	Gibt an, dass die Shell asynchrone Hinweise über Änderungen des Jobstatus sendet. Mit dem Standardwert werden Statusänderungen kurz vor der Anzeige der Shellvorgabeaufforderung angezeigt.
<i>path</i>	Gibt die Verzeichnisse an, in denen nach Befehlen zur Ausführung gesucht wird. Ein leeres Wort gibt das aktuelle Verzeichnis an. Ist die Variable <i>path</i> nicht gesetzt, werden nur vollständige Pfadnamen ausgeführt. Der Standardsuchpfad (ausgehend von der Datei <b>/etc/environment</b> , die beim Anmelden verwendet wird) lautet: <b>/usr/bin /etc /usr/sbin /usr/ucb /usr/bin/X11 /sbin</b> Eine Shell, in der weder das Flag <b>-c</b> noch das Flag <b>-t</b> angegeben wurde, schreibt normalerweise den Inhalt der in der Variablen <i>path</i> angegebenen Verzeichnisse in eine interne Hash-Tabelle, und zwar nach dem Lesen der Datei <b>.cshrc</b> und bei jedem Zurücksetzen der Variablen <i>path</i> . Wenn die neuen Befehle den Verzeichnissen hinzugefügt werden, während die Shell aktiv ist, müssen Sie den Befehl <b>rehash</b> verwenden. Andernfalls werden die Befehle unter Umständen nicht gefunden.

Eintrag	Beschreibung
<i>prompt</i>	Gibt die Zeichenfolge an, die angezeigt wird, bevor die einzelnen Befehle aus der Eingabe an einer interaktiven Workstation gelesen werden. Wenn die Zeichenfolge ein Ausrufezeichen ! enthält, wird dieses durch die aktuelle Ereignisnummer ersetzt. Falls das Ausrufezeichen (!) in einer Zeichenfolge enthalten ist, die in einfache oder doppelte Anführungszeichen eingeschlossen ist, muss dem Ausrufezeichen (!) ein Backslash (\) vorangestellt werden. Die Standardeingabeaufforderung für Benutzer mit Rootberechtigung ist %. Die Standardeingabeaufforderung des Benutzers mit Rootberechtigung ist #.
<i>savehist</i>	Gibt einen numerischen Wert an, der die Anzahl der Einträge in der Protokollliste steuert, die beim Abmelden in der Datei ~/.history gespeichert werden. Alle Befehle, auf die in dieser Anzahl von Ereignissen verwiesen wird, werden gespeichert. Während des Systemstarts liest die Shell die Datei ~/.history in die Protokollliste ein. Damit ist das Protokoll über die Anmeldung hinaus gespeichert. Sehr große Werte der Variablen <i>savehist</i> verlangsamen den Start der Shell.
<i>shell</i>	Gibt die Datei an, in der sich die C-Shell befindet. Diese Variable wird bei der Verzweigung von Shells verwendet, um Dateien zu interpretieren, in denen Ausführungsbits gesetzt sind, die aber vom System nicht ausgeführt werden können. Die Variable wird mit dem Pfadnamen der C-Shell initialisiert.
<i>status</i>	Gibt den Status an, der vom letzten Befehl zurückgegeben wurde. Wird der Befehl abnormal beendet, wird der Statusangabe 0200 hinzugefügt. Integrierte Befehle, die nicht erfolgreich ausgeführt wurden, geben den Exit-Status 1 zurück. Bei erfolgreich ausgeführten integrierten Befehlen wird der Status auf den Wert 0 gesetzt.
<i>time</i>	Steuert die automatische Ablaufsteuerung von Befehlen. Ist diese Variable gesetzt, zeigt jeder Befehl, der mehr als die angegebene Anzahl an CPU-Sekunden benötigt, nach seiner Ausführung eine Zeile mit den benutzten Ressourcen an. Nähere Informationen zu den Standardausgaben finden Sie in der Beschreibung des integrierten Befehls <b>time</b> .
<i>verbose</i>	Wird durch das Befehlszeilen-Flag <b>-v</b> gesetzt und bewirkt, dass die Wörter der einzelnen Befehle nach der Protokollsubstitution angezeigt werden.

## Jobsteuerung in der C-Shell:

Die Shell ordnet jedem Prozess eine Jobnummer zu. Die Shell führt eine Tabelle der aktuellen Jobs und ordnet ihnen kleine, ganze Zahlen mit normalem Wertebereich zu.

Wird ein Job mit einem Et-Zeichen (&) als Hintergrundjob gestartet, gibt die Shell eine Zeile wie die folgende aus:

```
[1] 1234
```

Die Zeile zeigt an, dass die Jobnummer 1 ist und der Job aus einem einzelnen Prozess besteht, der die Prozess-ID 1234 hat. Mit dem integrierten Befehl **jobs** kann die Tabelle der aktuellen Jobs angezeigt werden.

Ein Job, der im Hintergrund ausgeführt wird, steht in einer Konkurrenzsituation in Bezug auf die Eingabe, wenn er versucht, von der Workstation zu lesen. Hintergrundjobs können auch Ausgaben für die Workstation erzeugen, die mit der Ausgabe anderer Jobs verzahnt werden.

Sie können in der Shell auf mehrere Arten auf Jobs verweisen. Verwenden Sie das Prozentzeichen (%), um den Namen eines Jobs anzugeben. Als Name gilt die Jobnummer oder der Befehlsname, mit dem der Job gestartet wurde, falls dieser Name eindeutig ist. Wenn beispielsweise ein **make**-Prozess als Job 1 ausgeführt wird, können Sie ihn mit %1 angeben. Sie können auch die Angabe %make verwenden, falls nur ein Job, dessen Ausführung ausgesetzt wurde, mit einem Namen vorhanden ist, der mit der Zeichenfolge **make** beginnt. Außerdem können Sie folgende Angaben verwenden:

```
??Zeichenfolge
```

Hiermit geben Sie einen Job an, dessen Name die Variable *Zeichenfolge* enthält. Diese Angabe ist nur verwendbar, wenn nur ein solcher Job existiert.

Die Shell erkennt sofort, wenn ein Prozess seinen Status ändert. Ist ein Job so blockiert, dass keine weitere Ausführung möglich ist, sendet die Shell eine Nachricht an die Workstation. Diese Nachricht wird erst



angezeigt, nachdem Sie die Eingabetaste gedrückt haben. Ist jedoch die Shellvariable *notify* gesetzt, gibt die Shell sofort eine Nachricht aus, die die Statusänderung von Hintergrundjobs anzeigt. Der integrierte Befehl **notify** ist zu verwenden, um einen Einzelprozess zu markieren, so dass dessen Statusänderungen sofort gemeldet werden. Standardmäßig markiert der Befehl **notify** den aktuellen Prozess.

### Liste der integrierten C-Shellbefehle:

Im Folgenden sind die integrierten C-Shellbefehle aufgelistet.

Eintrag	Beschreibung
@	Zeigt den Wert angegebener Shellvariablen an.
alias	Zeigt den angegebenen Aliasnamen oder alle Aliasnamen an.
bg	Setzt die Ausführung des aktuellen bzw. angegebenen Jobs im Hintergrund fort.
break	Nimmt die Ausführung nach dem Ende der nächsten Schleife <b>foreach</b> oder <b>while</b> wieder auf.
breaksw	Verlässt eine <b>switch</b> -Anweisung.
case	Definiert einen Kennsatz in einem Befehl <b>switch</b> .
cd	Wechselt vom aktuellen Verzeichnis in das angegebene Verzeichnis.
chdir	Wechselt vom aktuellen Verzeichnis in das angegebene Verzeichnis.
continue	Setzt die Ausführung mit dem Start des nächsten Durchlaufs einer <b>foreach</b> - oder <b>while</b> -Schleife fort.
default	Kennzeichnet den Standardfall einer <b>switch</b> -Anweisung.
dirs	Zeigt den Verzeichnisstapel an.
echo	Schreibt Zeichenfolgen in die Standardausgabe der Shell.
else	Führt die Befehle aus, die dem zweiten <b>else</b> in einer Befehlsfolge des Typs <b>if (Ausdruck) then ...else if (Ausdruck2) then ... else ... endif</b> folgen.
end	Bezeichnet das Ende einer Befehlsfolge, der der Befehl <b>foreach</b> vorangestellt ist.
endif	Führt die Befehle aus, die der zweiten Anweisung <b>then</b> in einer Befehlsfolge des Typs <b>if (Ausdruck) then ... else if (Ausdruck2) then ... else ... endif</b> folgen.
endsw	Kennzeichnet das Ende einer Befehlsfolge des Typs <b>switch (Zeichenfolge) case Zeichenfolge : ... breaksw default: ... breaksw endsw</b> . Diese Befehlsfolge gleicht nacheinander alle case-Fälle mit dem Wert der Variablen <i>Zeichenfolge</i> ab. Die Ausführung des Scripts wird nach dem Ende der Anweisung <b>endsw</b> fortgesetzt, wenn ein Befehl <b>breaksw</b> ausgeführt wird oder keine Fallübereinstimmung gefunden wird und kein Standardwert existiert.
eval	Liest die Variablen als Eingabe für die Shell und führt die entsprechenden Befehle im Kontext der aktuellen Shell aus.
exec	Führt an Stelle der Shell den angegebenen Befehl aus.
exit	Verlässt die Shell entweder mit dem Wert der Shellvariablen oder mit dem Wert des angegebenen Ausdrucks.
fg	Stellt den aktuellen oder angegebene Jobs in den Vordergrund und setzt deren Ausführung fort, falls sie gestoppt wurden.
foreach	Setzt nacheinander die Variable <i>Name</i> auf jeden in der Variablen <i>Liste</i> angegebenen Eintrag und führt die Befehlsfolge zwischen der Anweisung <b>foreach</b> und der zugehörigen Anweisung <b>end</b> aus.
glob	Zeigt eine Liste mit dem Protokoll, den Variablen und Dateinamenerweiterungen an.
goto	Setzt die Ausführung nach einer angegebenen Zeile fort.
hashstat	Zeigt Statistiken an, die angeben, mit welchem Erfolg Befehle über die Hash-Tabelle gefunden wurden.
history	Zeigt die Protokollereignisliste an.
if	Führt einen angegebenen Befehl aus, wenn ein angegebener Ausdruck wahr ist.
jobs	Listet die aktiven Jobs auf.
kill	Sendet entweder das Abbruchsignal <b>TERM</b> oder das in der Variablen <i>Signal</i> angegebene Signal an den angegebenen Job oder Prozess.
limit	Begrenzt die Verwendung einer angegebenen Ressource durch den aktuellen Prozess und jeden von diesem erstellten Prozess.
login	Beendet eine Anmeldeshell und ersetzt sie durch ein Instanz des Befehls <i>/usr/sbin/login</i> .
logout	Beendet eine Anmeldeshell.
nice	Setzt die Priorität für Befehle, die in der Shell ausgeführt werden.
nohup	Bewirkt, dass HUP-Signale (Hangup) für den Rest einer Prozedur ignoriert werden.
notify	Weist die Shell an, den Benutzer asynchron zu benachrichtigen, wenn sich der Status des aktuellen oder angegebenen Jobs ändert.
onintr	Steuert die Aktion der Shell bei Unterbrechungen.
popd	Zeigt die Verzeichnisstapel an und wechselt zum neuen Hauptverzeichnis.
pushd	Tauscht Elemente des Verzeichnisstapels aus.
rehash	Bewirkt eine erneute Berechnung der internen Hash-Tabelle mit dem Inhalt der in der Shellpfadvariablen angegebenen Verzeichnisse.

<b>Eintrag</b>	<b>Beschreibung</b>
<b>repeat</b>	Führt den angegebenen Befehl, der denselben Einschränkungen wie der Befehl <b>if</b> unterliegt, so oft wie angegeben aus.
<b>set</b>	Zeigt die Werte aller Shellvariablen an.
<b>setenv</b>	Ändert den Wert der angegebenen Umgebungsvariablen.
<b>shift</b>	Verschiebt die angegebene Variable nach links.
<b>source</b>	Liest den in der Variablen <i>Name</i> angegebenen Befehl.
<b>stop</b>	Stoppt den aktuellen Job oder die angegebenen Jobs, die im Hintergrund ausgeführt werden.
<b>suspend</b>	Stoppt die Shell so, als wäre ein <b>STOP</b> empfangen worden.
<b>switch</b>	Startet eine Befehlsfolge des Typs <b>switch</b> ( <i>Zeichenfolge</i> ) <b>case</b> <i>Zeichenfolge</i> : ... <b>breaksw</b> <b>default</b> : ... <b>breaksw</b> <b>endsw</b> . Diese Befehlsfolge gleicht nacheinander alle <b>case</b> -Fälle mit dem Wert der Variablen <i>Zeichenfolge</i> ab. Wenn keine Fallübereinstimmung gefunden wird, bevor ein Standardfall gefunden wird, beginnt die Ausführung nach dem Standardfall.
<b>time</b>	Zeigt eine Übersicht der von der Shell und ihren Kindprozessen verwendeten Zeit an.
<b>umask</b>	Legt die Dateiberechtigungen fest.
<b>unalias</b>	Löscht alle Aliasnamen, die mit der Variablen <i>Muster</i> übereinstimmen.
<b>unhash</b>	Inaktiviert die Verwendung der internen Hash-Tabelle, um aktive Programm zu suchen.
<b>unlimit</b>	Entfernt Ressourcenbeschränkungen.
<b>unset</b>	Löscht alle Variablen, deren Name mit der Variablen <i>Muster</i> übereinstimmen.
<b>unsetenv</b>	Löscht alle Variablen aus der Umgebung, deren Namen mit der angegebenen Variablen <i>Muster</i> übereinstimmen.
<b>wait</b>	Wartet auf alle Hintergrundjobs.
<b>while</b>	Wertet die Befehle zwischen <b>while</b> und der zugehörigen Befehlsfolge <b>end</b> aus, solange der in der Variablen <i>Ausdruck</i> angegebene Ausdruck ungleich null ist.

Referenzinformationen:

### Korn-Shell

Die Befehle **ksh** und **stty**.

Die Korn-Shellbefehle **alias**, **cd**, **export**, **fc**, **getopts**, **read**, **set** und **typeset**.

Die Datei `/etc/passwd`.

### Bourne-Shell

Der Befehl **bsh** oder **rsh** und der Befehl **login**.

Der Bourne-Shellsonderbefehl **read**.

Die Subroutine **setuid**, die Subroutine **setgid**.

Die Gerätedatei `null`.

Die Datei `environment`, das Dateiformat `profile`.

### C-Shell

Die Befehle **csh** und **ed**.

Die integrierten C-Shellbefehle **alias**, **unalias**, **jobs**, **notify** und **set**.

### Zugehörige Konzepte:

„C-Shell“ auf Seite 288

Die C-Shell ist ein interaktiver Befehlsinterpreter und verwendet eine befehlsorientierte Programmiersprache. Die verwendete Syntax ist der Programmiersprache C ähnlich.

„Integrierte C-Shellbefehle“ auf Seite 299

Integrierte Befehle werden innerhalb der Shell ausgeführt. Tritt ein integrierter Befehl als Komponente einer Befehlskette (mit Ausnahme der letzten) auf, wird der Befehl in einer Subshell ausgeführt.

### Signalverarbeitung in der C-Shell:

Die C-Shell ignoriert in der Regel `quit`-Signale. Jobs, die im Hintergrund ausgeführt werden, sind von Signalen, die mit der Tastatur ausgelöst werden (**INTERRUPT**, **QUIT** und **HANGUP**) nicht betroffen.

Andere Signale haben die Werte, die die Shell vom Elternprozess übernommen hat. Die Verarbeitung der Signale **INTERRUPT** und **TERMINATE** durch die Shell in Shellprozeduren kann mit **onintr** gesteuert werden. Anmeldeshells erfassen oder ignorieren **TERMINATE**-Signale abhängig von ihrer Einstellung. Andere Shells als Anmeldeshells übergeben **TERMINATE**-Signale an den Kindprozess. Auf keinen Fall sind **INTERRUPT**-Signale erlaubt, wenn eine Anmeldeshell die Datei `.logout` liest.

### C-Shellbefehle:

Ein einfacher Befehl ist eine Folge von Wörtern, die durch Leerzeichen oder Tabulatorzeichen getrennt sind. Ein *Wort* ist eine Folge von Zeichen und/oder Zahlen, die keine Leerzeichen ohne Anführungszeichen enthält.

Zusätzlich bilden auch die folgenden Zeichen und doppelten Zeichen einzelne Wörter, wenn sie als Befehlstrennzeichen oder Abschlusszeichen verwendet werden:

&		;	>
&&		<<	>>
<	>	(	)

Diese Sonderzeichen können Teil von anderen Wörtern sein. Wenn ihnen ein Backslash (\) vorangestellt wird, interpretiert die Shell sie nicht als Sonderzeichen. Zeichenfolgen, die in einfache oder doppelte Anführungszeichen ( ' ' oder " ") oder in umgekehrte Anführungszeichen ( ` ` ) gesetzt sind, können ebenfalls als Teile von Wörtern auftreten. Leerzeichen, Tabulatorzeichen und Sonderzeichen bilden keine eigenen Wörter, wenn sie in diese Anführungszeichen gesetzt werden. Sie können auch ein Zeilenvorschubzeichen in diese Anführungszeichen einschließen, wenn Sie ihnen einen Backslash (\) voranstellen.

Das erste Wort in einer einfachen Befehlsfolge (mit der Nummer 0) gibt gewöhnlich den Namen des Befehls an. Alle übrigen Wörter werden mit wenigen Ausnahmen an diesen Befehl übergeben. Wenn der Befehl eine ausführbare Datei angibt, die ein kompiliertes Programm ist, führt die Shell das Programm sofort aus. Ist die Datei zwar als ausführbar markiert, aber kein kompiliertes Programm, behandelt die Shell sie wie ein Shell-Script. In diesem Fall erzeugt die Shell eine weitere Instanz von sich selbst (eine Subshell), um die Datei zu lesen und die darin enthaltenen Befehle auszuführen.

#### Integrierte C-Shellbefehle:

Integrierte Befehle werden innerhalb der Shell ausgeführt. Tritt ein integrierter Befehl als Komponente einer Befehlskette (mit Ausnahme der letzten) auf, wird der Befehl in einer Subshell ausgeführt.

**Anmerkung:** Wird ein Befehl an der Eingabeaufforderung der C-Shell eingegeben, sucht das System zuerst nach einem integrierten Befehl. Ist kein integrierter Befehl vorhanden, durchsucht das System die in der Shellvariablen *path* angegebenen Verzeichnisse nach einem Befehl auf Systemebene. Einige integrierte Befehle der C-Shell haben denselben Namen wie Betriebssystembefehle. Allerdings arbeiten diese Befehle nicht unbedingt auf dieselbe Weise. Nähere Informationen zur Funktionsweise eines Befehls finden Sie in der zugehörigen Befehlsbeschreibung.

Wenn Sie ein Shell-Script in der Shell ausführen und die erste Zeile des Shell-Scripts mit `#!/Shell Pfadname` beginnt, führt die C-Shell zur Verarbeitung des Scripts die Shell aus, die im Kommentar angegeben ist. Andernfalls ruft sie die Standardshell auf (d. h. die Shell, die mit `/usr/bin/sh` verbunden ist). Bei der Ausführung in der Standardshell werden integrierte C-Shellbefehle möglicherweise nicht erkannt. Zur Ausführung von C-Shellbefehlen müssen Sie die erste Zeile des Scripts wie folgt ändern: `#!/usr/bin/csh`.

#### Zugehörige Verweise:

„Liste der integrierten C-Shellbefehle“ auf Seite 297  
Im Folgenden sind die integrierten C-Shellbefehle aufgelistet.

## Beschreibungen der C-Shellbefehle:

In der C-Shell sind die folgenden integrierten Befehle verfügbar.

Eintrag	Beschreibung
<b>alias</b> [ <i>Name</i> [ <i>Wortliste</i> ]]	Zeigt alle Aliasnamen an, wenn keine Parameter angegeben werden. Andernfalls wird der Aliasname für den angegebenen <i>Namen</i> angezeigt. Wird die Variable <i>Wortliste</i> angegeben, ordnet der Befehl den Wert der <i>Wortliste</i> dem <i>Aliasnamen</i> zu. Der angegebene <i>Name</i> (Aliasname) darf weder <b>alias</b> noch <b>unalias</b> sein.
<b>bg</b> [% <i>Job</i> ...]	Stellt den aktuellen Job bzw. den mit <i>Job</i> angegebenen Job in den Hintergrund und setzt die Ausführung des Jobs fort, wenn dieser gestoppt war.
<b>break</b>	Nimmt die Ausführung nach der Anweisung <b>end</b> der nächsten Schleife <b>foreach</b> oder <b>while</b> wieder auf.
<b>breaksw</b>	Unterbricht beim Befehl <b>switch</b> und nimmt die Ausführung nach dem Befehl <b>endsw</b> wieder auf.
<b>case</b> <i>Kennsatz</i> :	Definiert einen <i>Kennsatz</i> in einem Befehl <b>switch</b> .
<b>cd</b> [ <i>Name</i> ]	Äquivalent zum Befehl <b>chdir</b> (siehe folgende Beschreibung).
<b>chdir</b> [ <i>Name</i> ]	Wechselt vom aktuellen Verzeichnis in das in der Variablen <i>Name</i> angegebene Verzeichnis. Wird kein <i>Name</i> angegeben, wird mit dem Befehl in das Ausgangsverzeichnis gewechselt. Wenn der Wert der Variablen <i>Name</i> kein Unterverzeichnis des aktuellen Verzeichnisses ist und nicht mit <i>/</i> , <i>./</i> oder <i>../</i> beginnt, durchsucht die Shell jede Komponente der Shellvariablen <i>cdpath</i> nach einem Unterverzeichnis, das der Variablen <i>Name</i> entspricht. Falls die Variable <i>Name</i> eine Shellvariable mit einem Wert ist, der mit einem Schrägstrich ( <i>/</i> ) beginnt, prüft die Shell, ob es sich um ein Verzeichnis handelt. Der Befehl <b>chdir</b> entspricht dem Befehl <b>cd</b> .
<b>continue</b>	Setzt die Ausführung nach der Anweisung <b>end</b> der nächsten Schleife <b>while</b> oder <b>foreach</b> fort.
<b>default</b> :	Kennzeichnet den Standardfall (default) in einer Anweisung <b>switch</b> . Der Standardwert <b>default</b> sollte nach allen anderen <i>Kennsätzen</i> <b>case</b> angegeben werden.
<b>dirs</b>	Zeigt den Verzeichnisstapel an.
<b>echo</b>	Schreibt Zeichenfolgen in die Standardausgabe der Shell.
<b>else</b>	Führt die Befehle aus, die der zweiten Anweisung <b>else</b> in einer Befehlsfolge des Typs <b>if</b> ( <i>Ausdruck</i> ) <b>then</b> ... <b>else if</b> ( <i>Ausdruck2</i> ) <b>then</b> ... <b>else</b> ... <b>endif</b> folgen. <b>Anmerkung:</b> Die Anweisung <b>else</b> ist der in der <b>csh</b> integrierte Befehl bei Verwendung von <b>if</b> ( <i>Ausdruck</i> ) <b>then</b> .. <b>else</b> ... <b>endif</b> . Wenn der ( <i>Ausdruck</i> ) wahr ist, werden die Befehle bis zur Anweisung <b>else</b> ausgeführt. Ist der ( <i>Ausdruck</i> ) falsch, werden die Befehle zwischen den Anweisungen <b>else</b> und <b>endif</b> ausgeführt. Angaben, die in einzelne Anführungszeichen eingeschlossen sind, werden buchstabengetreu verwendet und nicht interpretiert.
<b>end</b>	Setzt nacheinander die Variable <i>Name</i> auf jeden in der Variablen <i>Liste</i> angegebenen Eintrag und führt die <i>Befehlsfolge</i> zwischen der Anweisung <b>foreach</b> und der zugehörigen Anweisung <b>end</b> aus. Die Anweisungen <b>foreach</b> und <b>end</b> müssen in separaten Zeilen stehen.  Verwendet die Anweisung <b>continue</b> , um die Schleife fortzusetzen, und die Anweisung <b>break</b> , um die Schleife vorzeitig zu beenden. Wird der Befehl <b>foreach</b> vom Terminal gelesen, gibt die C-Shell die Eingabeaufforderung ? (Fragezeichen) aus, damit <i>Befehle</i> eingegeben werden können. Befehle innerhalb von Schleifen, die an der Eingabeaufforderung ? eingegeben werden, werden nicht in die Protokollliste gestellt.
<b>endif</b>	Wenn die Variable <i>Ausdruck</i> wahr ist, werden die <i>Befehle</i> ausgeführt, die der ersten Anweisung <b>then</b> folgen. Ist <b>else if</b> <i>Ausdruck2</i> wahr, werden die <i>Befehle</i> ausgeführt, die der zweiten Anweisung <b>then</b> folgen. Wenn <b>else if</b> <i>Ausdruck2</i> falsch ist, werden die <i>Befehle</i> nach der Anweisung <b>else</b> ausgeführt. Es kann eine beliebige Anzahl von <b>else if</b> -Paaren angegeben werden. Es ist nur eine Anweisung <b>endif</b> erforderlich. Das Segment <b>else</b> ist optional. Die Wörter <b>else</b> und <b>endif</b> können nur am Anfang von Eingabezeilen verwendet werden. Das Segment <b>if</b> muss allein in einer Eingabezeile oder hinter einem Befehl <b>else</b> stehen.

Eintrag	Beschreibung
<b>endsw</b>	Vergleicht nacheinander alle <b>case</b> -Fälle mit dem Wert der Variablen <i>Zeichenfolge</i> . Die <i>Zeichenfolge</i> wird zuerst mit dem Befehl und Dateinamen erweitert. Verwenden Sie die Platzhalterzeichen *, ? und [ . . . ] in den <b>case</b> -Kensätzen, die durch Variablen ersetzt werden. Wird für keinen der Fälle eine Übereinstimmung gefunden, bevor ein Standardfall ( <b>default</b> ) gefunden wird, beginnt die Ausführung nach dem <b>Standardfall</b> . <b>case</b> und <b>default</b> müssen am Anfang der Zeile stehen. Der Befehl <b>breaksw</b> veranlasst die Fortsetzung der Befehlsausführung nach dem Befehl <b>endsw</b> . Andernfalls werden die nachfolgenden Fälle ( <b>case</b> und <b>default</b> ) wie in der Programmiersprache C nacheinander abgearbeitet. Falls keine Fallübereinstimmung gefunden wird und kein Standardfall ( <b>default</b> ) angegeben ist, wird die Ausführung nach dem Befehl <b>endsw</b> fortgesetzt.
<b>eval</b> <i>Parameter . . .</i>	Liest den Wert der Variablen <i>Parameter</i> als Eingabe für die Shell und führt den bzw. die entsprechenden Befehle im Kontext der aktuellen Shell aus. Mit diesem Befehl werden Befehle ausgeführt, die aufgrund einer Befehls- oder Variablensubstitution generiert wurden, da die Syntaxanalyse vor den Substitutionen erfolgt.
<b>exec</b> <i>Befehl</i>	Führt anstelle der Shell den angegebenen <i>Befehl</i> aus.
<b>exit</b> ( <i>Ausdruck</i> )	Beendet die Shell entweder mit dem Wert der Shellvariablen <i>status</i> (falls kein <i>Ausdruck</i> angegeben wurde) oder mit dem Wert des angegebenen <i>Ausdrucks</i> .
<b>fg</b> [% <i>Job</i> ...]	Stellt den aktuellen Job bzw. den für <i>Job</i> angegebenen Job in den Vordergrund und setzt die Ausführung des Jobs fort, wenn dieser gestoppt war.
<b>foreach</b> <i>Name (Liste) Befehl. . .</i>	Setzt nacheinander die Variable <i>Name</i> auf jeden in der Variablen <i>Liste</i> angegebenen Eintrag und führt die Befehlsfolge zwischen der Anweisung <b>foreach</b> und der zugehörigen Anweisung <b>end</b> aus.
<b>glob</b> <i>Liste</i>	Zeigt die <i>Liste</i> unter Verwendung von Protokoll, Variablen und Dateinamenerweiterung an. Setzt ein Nullzeichen zwischen Wörter und enthält am Ende keinen Wagenrücklauf.
<b>goto</b> <i>Wort</i>	Setzt die Ausführung nach der in der Variablen <i>Wort</i> angegebenen Zeile fort. Das angegebene <i>Wort</i> ist ein Dateiname und ein Befehl, die erweitert wurden, um eine <i>Zeichenfolge</i> in dem Format zu bilden, das in der Variablen <i>Kensatz</i> : angegeben wurde. Die Shell rekonstruiert die Eingabe so weit wie möglich und sucht nach einer Zeile im Format <i>Kensatz</i> :, dem Leerzeichen oder Tabulatorzeichen vorangestellt sind.
<b>hashstat</b>	Zeigt Statistiken an, die angeben, mit welchem Erfolg Befehle über die Hash-Tabelle gefunden wurden.
<b>history</b> [-r   -h] [ <i>n</i> ]	Zeigt die Protokollereignisliste an. Die ältesten Ereignisse werden zuerst angezeigt. Wird eine Anzahl <i>n</i> angegeben, wird nur die angegebene Anzahl der jüngsten Ereignisse angezeigt. Mit dem Flag <b>-r</b> wird die Reihenfolge, in der die Ereignisse angezeigt werden, umgekehrt, so dass die jüngsten Ereignisse zuerst angezeigt werden. Mit dem Flag <b>-h</b> wird die Protokollliste ohne führende Nummern angezeigt. Dieses Flag ist zu verwenden, um Dateien zu erzeugen, die für die Verwendung mit dem Flag <b>-h</b> des Befehls <b>source</b> geeignet sind.
<b>if</b> ( <i>Ausdruck</i> ) <i>Befehl</i>	Führt den angegebenen <i>Befehl</i> (einschließlich seiner Argumente) aus, wenn der angegebene <i>Ausdruck</i> wahr ist. Die Variablensubstitution für die Variable <i>Befehl</i> findet zu einem früheren Zeitpunkt statt, d. h., zur gleichen Zeit wie für den Rest der Anweisung <b>if</b> . Bei dem angegebenen <i>Befehl</i> muss es sich um einen einfachen Befehl handeln (und nicht um eine Befehlskette, Befehlsliste oder eine Klammerbefehlsliste). <b>Anmerkung:</b> Die Umleitung von Eingabe und Ausgabe wird auch dann ausgeführt, wenn die Variable <i>Ausdruck</i> falsch ist und der <i>Befehl</i> nicht ausgeführt wird.
<b>jobs</b> [-l]	Listet die aktiven Jobs auf. Wenn Sie das Flag <b>-l</b> (Kleinbuchstabe <i>l</i> ) mit dem Befehl <b>jobs</b> angeben, werden zusätzlich zu der Jobnummer und dem Jobnamen die Prozess-IDs angezeigt.
<b>kill</b> -l   [[- <i>Signal</i> ] % <i>Job</i> ...   <i>PID</i> ...]	Sendet entweder das Abbruchsignal <b>TERM</b> oder das in der Variablen <i>Signal</i> angegebene Signal an den angegebenen <i>Job</i> oder die angegebene <i>PID</i> (Prozess-ID). Signale werden entweder mit der Nummer oder mit dem Namen angegeben (wie in der Datei <i>/usr/include/sys/signal.h</i> definiert, nur ohne das Präfix <b>SIG</b> ). Wenn Sie das Flag <b>-l</b> (Kleinbuchstabe <i>l</i> ) angeben, werden die Signalnamen aufgelistet.

Eintrag	Beschreibung
<b>limit [-h] [Ressource [Max-Verwendung]]</b>	<p>Begrenzt die Verwendung der angegebenen Ressource durch den aktuellen Prozess und jeden von diesem erstellten Prozess. Grenzwerte für Prozessressourcen werden in der Datei <code>/etc/security/limits</code> definiert. Steuerbare Ressourcen sind CPU-Zeit, Dateigröße, Datengröße, Kernspeicherauszugsgröße und Speicherbelegung. Die maximal zulässigen Werte für diese Ressourcen werden mit dem Befehl <b>mkuser</b> festgelegt, wenn der Benutzer dem System hinzugefügt wird. Mit dem Befehl <b>chuser</b> können sie geändert werden.</p> <p>Grenzwerte können in die Kategorien veränderlich oder fest eingeteilt werden. Der Benutzer kann die veränderlichen Grenzwerte bis auf den durch die festen Grenzwerte festgelegten Wert erhöhen. Um einen veränderlichen Grenzwert über den festen Grenzwert hinaus zu erhöhen oder um feste Grenzwerte zu verändern, muss die Rootberechtigung vorliegen. Mit dem Flag <b>-h</b> werden die festen Grenzwerte anstelle der veränderlichen Grenzwerte angezeigt.</p> <p>Ist der Parameter <i>Max-Verwendung</i> nicht angegeben, wird mit dem Befehl <b>limit</b> der aktuelle Grenzwert für die angegebene Ressource angezeigt. Ist der Parameter <i>Ressource</i> nicht angegeben, werden mit dem Befehl <b>limit</b> die aktuellen Grenzwerte für alle Ressourcen angezeigt. Weitere Informationen zu den Ressourcen, die mit dem Unterbefehl <b>limit</b> gesteuert werden, finden Sie in den Beschreibungen der Subroutinen <b>getrlimit</b>, <b>setrlimit</b> und <b>vlimit</b> in der Veröffentlichung <i>Technical Reference: Base Operating System and Extensions, Volume 1</i>.</p> <p>Der Parameter <i>Max-Verwendung</i> für die CPU-Zeit wird im Format <i>hh:mm:ss</i> angegeben. Der Parameter <i>Max-Verwendung</i> für andere Ressourcen wird als Gleitkommazahl oder als ganze Zahl angegeben, der wahlweise ein Maßstabsfaktor folgen kann. Der Maßstabsfaktor ist <i>k</i> bzw. Kilobyte (1024 Byte), <i>m</i> bzw. Megabyte oder <i>b</i> bzw. Blöcke (die Einheiten, die von der Subroutine <b>ulimit</b> verwendet werden. Lesen Sie hierzu die Beschreibung in der Veröffentlichung <i>Technical Reference: Base Operating System and Extensions, Volume 2</i>). Wird kein Maßstabsfaktor angegeben, wird <i>k</i> für alle Ressourcen angenommen. Sowohl für Ressourcennamen als auch Maßstabsfaktoren genügen eindeutige Präfixe zu den Namen.</p> <p><b>Anmerkung:</b> Mit diesem Befehl wird der verfügbare physische Speicher (Speicherbelegung) für Prozesse nur dann begrenzt, wenn durch andere aktive Prozesse eine Konkurrenzsituation in Bezug auf den Systemspeicher besteht.</p>
<b>login</b>	<p>Beendet eine Anmeldeshell und ersetzt sie durch eine Instanz des Befehls <code>/usr/bin/login</code>. Dies ist eine Möglichkeit der Abmeldung (aus Gründen der Kompatibilität mit den Befehlen <b>ksh</b> und <b>bsh</b>).</p>
<b>logout</b>	<p>Beendet eine Anmeldeshell. Dieser Befehl muss verwendet werden, wenn die Option <code>ignoreeof</code> gesetzt ist.</p>
<b>nice [+n] [Befehl]</b>	<p>Setzt die Priorität von Befehlen, die in dieser Shell ausgeführt werden, auf 24, falls keine Werte angegeben wurden. Bei Angabe des Flags <i>+n</i> wird die Priorität plus der angegebenen Zahl gesetzt. Bei Angabe des Flags <i>+n</i> und eines <i>Befehls</i> wird der <i>Befehl</i> mit der Priorität 24 plus der angegebenen Zahl ausgeführt. Wenn Sie Rootberechtigung besitzen, können Sie die Anweisung <b>nice</b> mit einer negativen Zahl angeben. Der <i>Befehl</i> wird immer in einer Subshell ausgeführt, und es gelten die in einfachen <i>if</i>-Anweisungen festgelegten Einschränkungen für Befehle.</p>
<b>nohup [Befehl]</b>	<p>Bewirkt, dass <b>Hangups</b> für den Rest des Scripts ignoriert werden, wenn kein <i>Befehl</i> angegeben ist. Wird ein <i>Befehl</i> angegeben, wird dieser <i>Befehl</i> ausgeführt, wobei <b>Hangups</b> ignoriert werden. Um eine Befehlskette oder Befehlsliste auszuführen, die Befehlskette bzw. Befehlsliste in ein Shell-Script stellen, dem Script Ausführungsberechtigung erteilen und als Wert für die Variable <i>Befehl</i> verwenden. Alle Prozesse, die durch die Angabe eines <i>Et</i>-Zeichens (&amp;) als Hintergrundprozess ausgeführt werden, sind während des Abmeldevorgangs wirksam gegen <b>Hangup</b>-Signale geschützt. Die Prozesse sind jedoch weiterhin von explizit gesendeten <b>Hangup</b>-Signalen betroffen, sofern die Anweisung <b>nohup</b> nicht verwendet wird.</p>
<b>notify [%Job...]</b>	<p>Weist die Shell an, den Benutzer asynchron zu benachrichtigen, wenn sich der Status des aktuellen oder angegebenen <i>Jobs</i> ändert. Normalerweise gibt die Shell Nachrichten kurz vor Ausgabe der Shellvingabeaufforderung aus. Diese Funktion wird automatisch ausgeführt, wenn die Shellvariable <i>notify</i> gesetzt ist.</p>

Eintrag	Beschreibung
<b>onintr</b> [-   <i>Kennsatz</i> ]	Steuert die Aktion der Shell bei Unterbrechungen. Werden keine Argumente angegeben, wird die Standardaktion der Shell bei Unterbrechungen wiederhergestellt, die das Shell-Script beendet oder zur Befehleingabeebene zurückkehrt. Bei Angabe eines Flags - werden alle Unterbrechungen ignoriert. Bei Angabe von <i>Kennsatz</i> wird die Shell angewiesen, die Anweisung <code>goto <i>Kennsatz</i></code> auszuführen, wenn sie eine Unterbrechung empfängt oder wenn ein untergeordneter Prozess aufgrund einer Unterbrechung beendet wird. In jedem Fall verlieren alle Formate der Anweisung <b>onintr</b> ihre Bedeutung, wenn die Shell im Hintergrund ausgeführt wird und Unterbrechungen ignoriert werden. Unterbrechungen werden von der Shell und allen aufgerufenen Befehlen weiterhin ignoriert.
<b>popd</b> [+ <i>n</i> ]	Zeigt die Verzeichnisstapel an und wechselt zum neuen Hauptverzeichnis. Wenn Sie eine Variable + <i>n</i> angeben, verwirft der Befehl den <i>n</i> -ten Eintrag im Stapel. Die Elemente des Verzeichnisstapels sind von oben nach unten, beginnend mit 0, nummeriert.
<b>pushd</b> [+ <i>n</i>   <i>Name</i> ]	Werden keine Argumente angegeben, werden die zwei obersten Elemente des Verzeichnisstapels ausgetauscht. Wird die Variable <i>Name</i> angegeben, wechselt der Befehl in das neue Verzeichnis und stellt das alte aktuelle Verzeichnis (wie in der Shellvariablen <i>cwd</i> angegeben) in den Verzeichnisstapel. Wenn Sie eine Variable + <i>n</i> angeben, verschiebt der Befehl die <i>n</i> -te Komponente des Verzeichnisstapels an den Anfang und wechselt dorthin. Die Elemente des Verzeichnisstapels sind von oben nach unten, beginnend mit 0, nummeriert.
<b>rehash</b>	Bewirkt eine erneute Berechnung der internen Hash-Tabelle mit dem Inhalt der in der Shellvariablen <i>path</i> angegebenen Verzeichnisse. Diese Aktion ist erforderlich, wenn den Verzeichnissen in der Shellvariablen <i>path</i> neue Befehle hinzugefügt werden, während der Benutzer angemeldet ist. Der Befehl <b>rehash</b> ist nur dann erforderlich, wenn den benutzereigenen Verzeichnissen Befehle hinzugefügt werden oder wenn der Inhalt eines der Systemverzeichnisse geändert wurde.
<b>repeat</b> <i>Anzahl</i> <i>Befehl</i>	Führt den angegebenen <i>Befehl</i> , der denselben Einschränkungen wie Befehle in einfachen <code>if</code> -Anweisungen unterliegt, so oft aus, wie im Parameter <i>Anzahl</i> angegeben ist. <b>Anmerkung:</b> Umleitungen der Eingabe und Ausgabe werden genau einmal ausgeführt, selbst wenn in der Variablen <i>Anzahl</i> der Wert 0 (null) angegeben ist.
<b>set</b> [[ <i>Name</i> [ <i>n</i> ] [= <i>Wort</i> ]]   [ <i>Name</i> = ( <i>Liste</i> )]	Zeigt die Werte aller Shellvariablen an, wenn keine Argumente angegeben werden. Variablen, deren Wert aus mehreren Wörtern besteht, werden als Wortliste in Klammern angezeigt. Wird nur <i>Name</i> angegeben, setzt die C-Shell die Variable <i>Name</i> auf die leere Zeichenfolge. Andernfalls wird die Variable <i>Name</i> auf den Wert der Variablen <i>Wort</i> oder auf die mit der Variablen <i>Liste</i> angegebenen Wortliste gesetzt. Wenn <i>n</i> angegeben ist, wird die <i>n</i> -te Komponente der Variablen <i>Name</i> auf den Wert der Variablen <i>Wort</i> gesetzt. Die <i>n</i> -te Komponente muss bereits vorhanden sein. In allen Fällen besteht der Wert aus erweiterten Befehlen und Dateinamen. Diese Argumente können wiederholt werden, um mehrere Werte in einem einzelnen Befehl <b>set</b> zu setzen. Die Variablenerweiterung für alle Argumente findet jedoch statt, bevor eine Einstellung erfolgt.
<b>setenv</b> <i>Name</i> <i>Wert</i>	Setzt den Wert der in der Variablen <i>Name</i> angegebenen Umgebungsvariablen auf <i>Wert</i> , eine einzelne Zeichenfolge. Die am häufigsten verwendeten Umgebungsvariablen <b>USER</b> , <b>TERM</b> , <b>HOME</b> und <b>PATH</b> werden automatisch in bzw. aus den C-Shellvariablen <i>user</i> , <i>term</i> , <i>home</i> und <i>path</i> im- bzw. exportiert. Für diese Variablen muss die Anweisung <b>setenv</b> nicht verwendet werden.
<b>shift</b> [ <i>Variable</i> ]	Verschiebt die Elemente der Shellvariablen <i>argv</i> oder der angegebenen <i>Variablen</i> nach links. Wenn die Shellvariable <i>argv</i> oder die angegebene <i>Variable</i> nicht gesetzt ist oder ihr Wert weniger als ein Wort enthält, tritt ein Fehler auf.
<b>source</b> [- <i>h</i> ] <i>Name</i>	Liest Befehle, die in der Datei <i>Name</i> angegeben sind. Der Befehl <b>source</b> kann verschachtelt werden. Falls der Befehl jedoch zu tief verschachtelt wird, stehen der Shell möglicherweise nicht genügend Dateideskriptoren zur Verfügung. Tritt ein Fehler in einem Befehl <b>source</b> auf einer Ebene auf, werden alle verschachtelten Befehle <b>source</b> beendet. Normalerweise werden Eingaben, die während der Befehle <b>source</b> erfolgen, nicht in die Protokollliste gestellt. Mit dem Flag <b>-h</b> können die Befehle in die Protokollliste gestellt werden, ohne ausgeführt zu werden.
<b>stop</b> [% <i>Job</i> ...]	Stoppt den aktuellen Job oder den angegebenen <i>Job</i> , der im Hintergrund ausgeführt wird.
<b>suspend</b>	Stoppt die Shell so, als wäre ein <b>STOP</b> empfangen worden.

Eintrag	Beschreibung
<b>switch</b> ( <i>Zeichenfolge</i> )	Startet eine Befehlsfolge des Typs <b>switch</b> ( <i>Zeichenfolge</i> ) <b>case</b> <i>Zeichenfolge</i> : ... <b>breaksw default</b> : ... <b>breaksw endsw</b> . Diese Befehlsfolge gleicht nacheinander alle case-Fälle mit dem Wert der Variablen <i>Zeichenfolge</i> ab. Wenn keine Fallübereinstimmung gefunden wird, bevor ein Standardfall gefunden wird, beginnt die Ausführung nach dem Standardfall.
<b>time</b> [ <i>Befehl</i> ]	<p>Mit dem Befehl <b>time</b> wird die automatische Ablaufsteuerung von Befehlen gesteuert. Wird die Variable <i>Befehl</i> nicht angegeben, zeigt der Befehl <b>time</b> die Gesamtzeit an, die von der Shell und ihren untergeordneten Prozessen verwendet wurde. Wird über die Variable <i>Befehl</i> ein Befehl angegeben, ist er zeitverzögert. Die Shell zeigt dann eine Zeitübersicht an, wie sie bei der Shellvariablen <b>time</b> beschrieben wird. Falls erforderlich, wird eine zusätzliche Shell erstellt, um die Zeitstatistik nach Beendigung des Befehls anzuzeigen.</p> <p>Im folgenden Beispiel wird <b>time</b> mit dem Befehl <b>sleep</b> verwendet:</p> <pre>time sleep</pre> <p>Die Ausgabe des Befehls sieht in etwa wie folgt aus:</p> <pre>0.0u 0.0s 0:00 100% 44+4k 0+0io 0pf+0w</pre> <p>Die folgenden Ausgabefelder sind vorhanden:</p> <p><b>Eins</b> CPU-Zeit (in Sekunden) für den Benutzerprozess</p> <p><b>Zwei</b> CPU-Zeit (in Sekunden), die vom Kernel für den Benutzerprozess benötigt wurde</p> <p><b>Drei</b> Abgelaufene Zeit (Uhrzeit) für den Befehl</p> <p><b>Vier</b> Gesamt-CPU-Zeit plus Systemzeit, ausgedrückt als Prozentsatz der abgelaufenen Zeit</p> <p><b>Fünf</b> Durchschnittlicher gemeinsam benutzter Speicherbereich plus durchschnittlicher nicht gemeinsam benutzter Datenraum in Kilobyte</p> <p><b>Sechs</b> Anzahl der Blockeingebe- und Blockausgabeoperationen</p> <p><b>Sieben</b> Fehlseitenbedingungen plus Anzahl der Auslagerungen</p>
<b>umask</b> [ <i>Wert</i> ]	Legt die Dateiberechtigungen fest. Beim Erstellen einer Datei werden mit diesem <i>Wert</i> zusammen mit den Berechtigungen des Erstellungsprozesses die Berechtigungen der Datei festgelegt. Der Standardwert ist 022. Wird kein <i>Wert</i> angegeben, wird die aktuelle Einstellung angezeigt.
<b>unalias</b> *  <i>Muster</i>	Löscht alle Aliasnamen, die mit der Variablen <i>Muster</i> übereinstimmen. Mit dem Befehl <b>unalias</b> * werden alle Aliasnamen gelöscht. Sind keine Aliasnamen vorhanden, führt dies nicht zu einem Fehler.
<b>unhash</b>	Inaktiviert die Verwendung der internen Hash-Tabelle, um aktive Programm zu suchen.
<b>unlimit</b> [-h][ <i>Ressource</i> ]	<p>Löscht die Grenzwerte für die Variable <i>Ressource</i>. Ist die Variable <i>Ressource</i> nicht angegeben, werden die Grenzwerte aller Ressourcen gelöscht. In der Beschreibung des Befehls <b>limit</b> ist eine Liste der Namen der <i>Ressourcen</i> enthalten.</p> <p>Mit dem Flag <b>-h</b> werden die entsprechenden festen Grenzwerte gelöscht. Nur Benutzer mit Rootberechtigung können feste Grenzwerte ändern.</p>
<b>unset</b> *  <i>Muster</i>	Löscht alle Variablen, deren Name mit der Variablen <i>Muster</i> übereinstimmen. Mit dem Befehl <b>unset</b> * werden alle Variablen gelöscht. Sind keine Variablen gesetzt, tritt kein Fehler auf.
<b>unsetenv</b> <i>Muster</i>	Löscht alle Variablen aus der Umgebung, deren Namen mit dem angegebenen <i>Muster</i> übereinstimmen. (Weitere Informationen sind im Abschnitt zum integrierten Befehl <b>setenv</b> enthalten.)
<b>wait</b>	Wartet auf alle Hintergrundjobs. Ist die Shell interaktiv, wird der Wartestatus durch eine Unterbrechung ( <b>INTERRUPT</b> ), normalerweise die Tastenkombination Strg-C, unterbrochen. Die Shell zeigt daraufhin die Namen und Jobnummern aller ausstehenden Jobs an.



Eintrag	Beschreibung
<b>while</b> ( <i>Ausdruck</i> ) <i>Befehl</i> . . . <b>end</b>	Wertet die <i>Befehle</i> zwischen der Anweisung <b>while</b> und der zugehörigen Anweisung <b>end</b> aus, solange der mit der Variablen <i>Ausdruck</i> angegebene Ausdruck ungleich null ist. Mit der Anweisung <b>break</b> wird die Schleife vorzeitig beendet und mit der Anweisung <b>continue</b> fortgesetzt. Die Anweisungen <b>while</b> und <b>end</b> müssen jeweils allein in der jeweiligen Eingabezeile stehen. Erfolgt die Eingabe an einem Terminal, werden nach <b>while</b> ( <i>Ausdruck</i> ) Eingabeaufforderungen wie bei der Anweisung <b>foreach</b> angezeigt.
@ [ <i>Name</i> [ <i>n</i> ] = <i>Ausdruck</i> ]	Zeigt die Werte aller Shellvariablen an, wenn keine Argumente angegeben werden. Andernfalls wird der in der Variablen <i>Name</i> angegebene Name auf den Wert der Variablen <i>Ausdruck</i> gesetzt. Falls der Ausdruck das Zeichen <, >, & oder   enthält, muss dieser Teil des Ausdrucks in runde Klammern gesetzt werden. Wenn <i>n</i> angegeben ist, wird die <i>n</i> -te Komponente der Variablen <i>Name</i> auf die Variable <i>Ausdruck</i> gesetzt. Die Variable <i>Name</i> und die zugehörige <i>n</i> -te Komponente müssen bereits vorhanden sein.  Operatoren der Programmiersprache C wie *= und += werden unterstützt. Das Leerzeichen, das die Variable <i>Name</i> vom Zuordnungsoperator trennt, ist optional. Leerzeichen sind jedoch erforderlich, um die Komponenten der Variablen <i>Ausdruck</i> zu trennen, die andernfalls als ein einziges Wort interpretiert würden. Besondere Suffixoperatoren wie ++ (doppeltes Pluszeichen) und -- (doppeltes Minuszeichen) erhöhen bzw. verringern jeweils den Wert der Variablen <i>Name</i> .

### Ausdrücke und Operatoren in der C-Shell:

Der integrierte Befehl @ und die Anweisungen **exit**, **if** und **while** akzeptieren Ausdrücke, die Operatoren enthalten, welche denen der Programmiersprache C ähnlich sind. Dabei gilt dieselbe Vorrangstellung.

Die folgenden Operatoren sind verfügbar:

Operator	Bedeutung
()	Änderung der Vorrangstellung
~	Komplement
!	Negation
*/ %	Multiplikation, Division, Modulo
+ -	Addition, Subtraktion
<< > >	Verschieben nach links, Verschieben nach rechts
<= >= < >	Vergleichsoperatoren
== != =~ !~	Zeichenfolgevergleich/Musterabgleich
&	Bitweises AND
^	Bitweises exklusives OR
	Bitweises exklusives OR
&&	Logisches AND
	Logisches OR

In der obigen Liste sind die Zeilen mit den Operatoren nach abnehmender Vorrangstellung von links nach rechts und von oben nach unten angeordnet.

**Anmerkung:** Die Operatoren + und - sind rechts-assoziativ. Die Auswertung von a + b - c wird beispielsweise folgendermaßen ausgeführt:

a + (b - c)

und nicht:

(a + b) - c

Die Operatoren ==, !=, =~ und !~ vergleichen ihre Argumente als Zeichenfolgen. Alle anderen Operatoren arbeiten mit Zahlen. Die Operatoren =~ und !~ gleichen den Operatoren == und != mit der Ausnahme, dass es sich bei der rechten Hälfte um ein *Muster* handelt, gegen das der linke Operand abgeglichen wird. Auf diese Weise kann die Anzahl der Anweisungen **switch** in Shellprozeduren verringert werden.

Die logischen Operatoren **or** (|) und **and** (&&) sind ebenfalls verfügbar. Sie können unter anderem, wie im folgenden Beispiel gezeigt, verwendet werden, um Zahlbereiche zu überprüfen:

```
if ($#argv > 2 && $#argv < 7) then
```

Im obigen Beispiel muss die Anzahl an Argumenten größer als 2 und kleiner als 7 sein.

Zeichenfolgen, die mit null (0) beginnen, werden als Oktalzahlen interpretiert. Nullargumente oder fehlende Argumente werden als null interpretiert. Alle Ausdrücke ergeben Zeichenfolgen, die Dezimalzahlen darstellen. Es ist zu beachten, dass zwei Komponenten eines Ausdrucks in demselben Wort auftreten können. Ausdruckskomponenten sollten nur dann von runden Klammern eingeschlossen sein, wenn sie in der Nähe von Ausdruckskomponenten stehen, die wichtig für den Parser sind (& | < > ( )).

Als Basisoperanden in Ausdrücken sind außerdem Befehlsausführungen, die in runde Klammern ( ( ) ) eingeschlossen sind, und Dateiabfragen im Format (- **Operator** *Dateiname*) verfügbar. Die gültigen Optionen für **Operator** sind im Folgenden aufgeführt:

Eintrag	Beschreibung
r	Lesezugriff
w	Schreibzugriff
x	Ausführungszugriff
e	Existenz
o	Eigentumsrecht
z	Größe null
f	Unverschlüsselte Datei
d	Verzeichnis

Bei dem angegebenen *Dateinamen* werden Befehle und Dateinamen erweitert und anschließend daraufhin überprüft, ob die angegebene Abhängigkeit mit dem realen Benutzer besteht. Ist der *Dateiname* nicht vorhanden oder kann nicht auf die Datei zugegriffen werden, wird für alle Abfragen `false(0)` zurückgegeben. Wird der Befehl erfolgreich ausgeführt, gibt die Abfrage den Wert `true(1)` zurück. Andernfalls, d. h., wenn der Befehl fehlschlägt, wird der Wert `false(0)` zurückgegeben. Sind zusätzliche detailliertere Statusinformationen erforderlich, kann der Befehl außerhalb eines Ausdrucks ausgeführt werden und anschließend die Shellvariable *Status* überprüft werden.

#### *Befehlssubstitution in der C-Shell:*

Bei der *Befehlssubstitution* führt die Shell den angegebenen Befehl aus und ersetzt den Befehl durch seine Ausgabe.

Zur Ausführung der Befehlssubstitution in der C-Shell muss der Befehl oder die Befehlsfolge in umgekehrte Anführungszeichen ( ` ` ) eingeschlossen werden. Die Shell teilt die Ausgabe des Befehls normalerweise in einzelne Wörter auf, und zwar an den Stellen, an denen Leerzeichen, Tabulatorzeichen und Zeilenvorschubzeichen vorkommen. Anschließend ersetzt sie den ursprünglichen Befehl durch dessen Ausgabe.

Im folgenden Beispiel zeigen die umgekehrten Anführungszeichen ( ` ` ) um den Befehl **date** an, dass die Ausgabe des Befehls ersetzt wird:

```
echo Aktuelles Datum und Zeit: `date`
```

Dieser Befehl könnte folgende Ausgabe haben:

```
Aktuelles Datum und Zeit: Mi Apr 8 13:52:14 1992
```

Die C-Shell führt eine selektive Befehlssubstitution bei den Argumenten von integrierten Shellbefehlen durch. Das bedeutet, dass die Teile von Ausdrücken, die nicht ausgewertet werden, nicht ersetzt werden.

Bei nicht integrierten Befehlen ersetzt die Shell den Befehlsnamen getrennt von der Argumentenliste. Die Substitution wird in einem Kindprozess der Haupt-Shell erst ausgeführt, nachdem die Shell die Umleitung der Eingabe und Ausgabe ausgeführt hat.

Ist eine Befehlsfolge in doppelte Anführungszeichen (" ") gesetzt, interpretiert die Shell ausschließlich Zeilenvorschubzeichen als Worttrennzeichen und behält somit Leerzeichen und Tabulatorzeichen innerhalb des Wortes bei. In allen Fällen erzwingt ein einzelnes abschließendes Zeilenvorschubzeichen kein neues Wort.

### Zugehörige Konzepte:

„Variablensubstitution in der C-Shell“ auf Seite 290

Die C-Shell verwaltet einen Satz von Variablen, von denen jede über eine Liste von null oder mehr Wörtern als Wert verfügt. Einige dieser Variablen werden von der Shell gesetzt, oder es wird auf sie verwiesen. Die Variable *argv* beispielsweise ist ein Abbild der Shellvariablenliste. Auf Wörter, die den Wert dieser Variablen enthalten, wird auf besondere Weise verwiesen.

*Ausführung nicht integrierter C-Shellbefehle:*

Stellt die C-Shell fest, dass es sich bei einem Befehl um einen nicht integrierten Shellbefehl handelt, versucht sie, ihn mit der Subroutine **execv** auszuführen.

Jedes Wort in der Shellvariablen *path* benennt ein Verzeichnis, von dem aus die Shell versucht, den Befehl auszuführen. Ist weder das Flag **-c** noch das Flag **-t** angegeben, setzt die Shell die Namen dieser Verzeichnisse in eine interne Hash-Tabelle. Die Shell versucht nur dann, die Subroutine **execv** in einem Verzeichnis aufzurufen, wenn die Möglichkeit besteht, dass sich der Befehl dort befindet. Wird dieser Mechanismus mit dem Befehl **unhash** ausgeschaltet oder wird das Flag **-c** oder **-t** für die Shell angegeben, verknüpft die Shell sie mit dem angegebenen Befehlsnamen, um einen Pfadnamen einer Datei zu bilden. Diesen Schritt führt die Shell auch für jede Verzeichniskomponente der Variablen *path* aus, die mit einem Schrägstrich (/) beginnt. Anschließend versucht die Shell den Befehl auszuführen.

Klammerbefehle werden immer in einer Subshell ausgeführt. Beispiel:

```
(cd ; pwd) ; pwd
```

Mit diesem Befehl wird das Ausgangsverzeichnis angezeigt, ohne das aktuelle Verzeichnis zu wechseln. Mit dem Befehl

```
cd ; pwd
```

wird jedoch vom aktuellen Verzeichnis in das Ausgangsverzeichnis gewechselt. Klammerbefehle werden meistens verwendet, um Auswirkungen des Befehls **chdir** auf die aktuelle Shell zu verhindern.

Hat die Datei Ausführungsberechtigung, handelt es sich aber nicht um eine vom System ausführbare Binärdatei, behandelt die Shell sie wie eine Datei mit Shellbefehlen und ruft eine neue Shell auf, um sie zu lesen.

Ist ein Aliasname für die Shell vorhanden, werden die Wörter der Aliasnamen der Argumentenliste vorgeschaltet, um den Shellbefehl zu bilden. Das erste Wort des Aliasnamens sollte der vollständige Pfadname der Shell sein.

### Protokollsubstitution in der C-Shell:

Mit der Protokollsubstitution können einzelne Wörter von vorherigen Befehlen geändert werden, um somit neue Befehle zu erstellen. Die Protokollsubstitution erleichtert die Wiederholung von Befehlen, die Wiederholung von Argumenten eines vorherigen Befehls im aktuellen Befehl oder die Berichtigung von Schreibfehlern im vorherigen Befehl mit geringem Eingabeaufwand.

Protokollsubstitutionen beginnen mit einem Ausrufezeichen (!) und können überall in der Befehlszeile auftreten, vorausgesetzt, sie sind nicht verschachtelt (d. h., eine Protokollsubstitution darf keine andere Protokollsubstitution enthalten). Sie können dem Ausrufezeichen (!) einen Backslash (\) voranstellen, damit die Sonderbedeutung des Ausrufezeichens ignoriert wird. Wenn Sie das Ausrufezeichen (!) vor ein Leerzeichen, ein Tabulatorzeichen, ein Zeilenvorschubzeichen, ein Gleichheitszeichen (=) oder eine linke runde Klammer (()) setzen, wird keine Protokollsubstitution durchgeführt.

Protokollsubstitutionen können auch stattfinden, wenn Sie eine Eingabezeile mit einem Winkelzeichen (^) beginnen. Die Shell meldet jede Eingabezeile, die Protokollsubstitutionen enthält, an die Workstation zurück, bevor die Zeile ausgeführt wird.

### Zugehörige Konzepte:

„Aliassubstitution in der C-Shell“ auf Seite 289

Ein *Aliasname* ist ein Name, der einem Befehl oder einer Befehlsfolge zugeordnet ist. In der C-Shell können Aliasnamen zugeordnet und wie Befehle verwendet werden. Die Shell verwaltet eine Liste der definierten Aliasnamen.

### Protokolllisten für die C-Shell:

In der Protokollliste werden die Befehle gespeichert, die die Shell aus der Befehlszeile liest und die aus mindestens einem Wort bestehen. Aus diesen gespeicherten Befehlen führt die Protokollsubstitution Wortfolgen erneut in den Eingabedatenstrom ein.

Mit der Shellvariablen *history* wird die Größe der Protokollliste gesteuert. Sie müssen die Shellvariable *history* in der Datei `.cshrc` oder in der Befehlszeile mit dem integrierten Befehl `set` setzen. Der vorherige Befehl wird unabhängig vom Wert der Variablen *history* beibehalten. Die Befehle in der Protokollliste sind sequenziell, beginnend mit 1, nummeriert. Der integrierte Befehl `history` erzeugt eine Ausgabe wie die folgende:

```
9 write michael
10 ed write.c
11 cat oldwrite.c
12 diff *write.c
```

Die Shell zeigt die Befehlsfolgen mit den entsprechenden Ereignisnummern an. Die Ereignisnummer erscheint links vom Befehl und verdeutlicht, wann der Befehl in Relation mit den anderen Befehlen im Protokoll eingegeben wurde. Es ist normalerweise nicht erforderlich, über die Ereignisnummern auf Ereignisse zu verweisen. Die aktuelle Ereignisnummer kann jedoch als Teil der Eingabeaufforderung des Systems angezeigt werden, indem ein Ausrufezeichen (!) in der Zeichenfolge für die Eingabeaufforderung gesetzt wird, die der Umgebungsvariablen *PROMPT* zugeordnet ist.

Ein vollständiger Protokollverweis enthält eine Ereignisspezifikation, eine Wortkennung und mindestens einen Änderungswert im folgenden allgemeinen Format:

```
Ereignis[.]Wort:Änderungswert[:Änderungswert] . . .
```

**Anmerkung:** Es kann nur ein Wort geändert werden. Eine Zeichenfolge mit Leerzeichen ist nicht zulässig.

Im vorherigen Beispiel für die Ausgabe eines Befehls `history` ist die aktuelle Ereignisnummer 13. Bei Verwendung dieses Beispiels verweisen die folgenden Formate auf vorherige Ereignisse:

Eintrag	Beschreibung
!10	Ereignisnummer 10.
!-2	Ereignisnummer 11 (das aktuelle Ereignis minus 2).
!d	Befehlswort, das mit d beginnt (Ereignisnummer 12).
!?mic?	Befehlswort, das die Zeichenfolge mic enthält (Ereignisnummer 9).

Mit diesen Formaten können die Wörter der angegebenen Ereignisse, durch einzelne Leerzeichen voneinander getrennt, ohne weitere Änderungen einfach erneut eingeführt werden. Als Sonderfall verweist das doppelte Ausrufezeichen (!! ) auf den vorherigen Befehl. Wird der Befehl !! allein in einer Eingabezeile eingegeben, wird der vorherige Befehl erneut ausgeführt.

#### Ereignisspezifikation für die C-Shell:

Um Wörter eines Ereignisses auszuwählen, muss der Ereignisspezifikation ein Doppelpunkt (:) und eine der folgenden Wortkennungen nachgestellt werden (die Wörter in der Eingabezeile sind sequenziell, beginnend mit 0, nummeriert).

Eintrag	Beschreibung
0	Erstes Wort (der Befehlsname)
n	n-tes Argument
^	Erstes Argument
\$	Letztes Argument
%	Wort, das der unmittelbar zuvor durchgeführten Suchoperation <i>?Zeichenfolge?</i> entspricht
x-y	Bereich von Wörtern, beginnend beim x-ten Wort bis hin zum y-ten Wort
-y	Bereich von Wörtern, beginnend beim ersten Wort (0) bis hin zum y-ten Wort
*	Vom ersten bis zum letzten Argument oder keine Auswahl, wenn das Ereignis nur ein Wort (den Befehlsnamen) enthält
x*	x-tes Argument bis zum letzten Argument
x-	Wie x*, aber das letzte Argument wird ausgelassen

Falls die Wortkennung mit einem ^, \$, \*, - oder % beginnt, können Sie den Doppelpunkt weglassen, der Ereignisspezifikation und Wortkennung trennt. Außerdem kann eine Folge der nachstehenden Änderungswerte nach der optionalen Wortkennung eingegeben werden. Jedem Änderungswert muss ein Doppelpunkt vorangestellt werden:

Eintrag	Beschreibung
h	Löscht die abschließende Pfadnamenserweiterung und lässt den Kopf stehen.
r	Löscht eine abschließende .xxx-Komponente und lässt den Rootnamen stehen.
e	Löscht alles außer der abschließenden .xxx-Erweiterung.
s/ <i>AltesWort</i> / <i>NeuesWort</i> /	Ersetzt den Wert der Variablen <i>AltesWort</i> durch den Wert der Variablen <i>NeuesWort</i> .

Die linke Seite einer Substitution ist kein Muster im Sinne einer Zeichenfolge, die von einem Editor erkannt wird. Sie ist vielmehr ein Wort, eine einzelne Einheit ohne Leerzeichen. Normalerweise wird das ursprüngliche Wort (*AltesWort*) durch einen Schrägstrich (/) von seinem Ersatzwert (*NeuesWort*) abgetrennt. Als Begrenzer kann jedoch ein beliebiges Zeichen verwendet werden. Im folgenden Beispiel ist es durch die Verwendung des Prozentzeichens % als Begrenzer möglich, einen Schrägstrich / in den Wörtern zu verwenden.

```
s%/home/meinedatei%/home/ihredatei%
```

Die Shell ersetzt ein Et-Zeichen (&) durch den Text von *AltesWort* in der Variablen *NeuesWort*. Im folgenden Beispiel wird /home/meinedatei zu /temp/home/meinedatei.

```
s%/home/meinedatei%/temp&%
```

Die Shell ersetzt ein leeres Wort in einer Substitution entweder durch die letzte Substitution oder durch die letzte Zeichenfolge, die in der kontextbezogenen Suchoperation *!?Zeichenfolge?* verwendet wurde.

Das abschließende Begrenzungszeichen (/) muss nicht angegeben werden, wenn unmittelbar danach ein Zeilenvorschubzeichen folgt. Die folgenden Änderungswerte stehen zur Begrenzung einer Protokollliste zur Verfügung:

Eintrag	Beschreibung
t	Löscht alle führenden Komponenten von Pfadnamen und lässt den Nachsatz stehen.
&	Wiederholt die vorherige Substitution.
g	Wendet die Änderung global an, d. h. für alle Vorkommen in jeder Zeile.
p	Zeigt den neuen Befehl an, führt ihn aber nicht aus.
q	Setzt die ersetzten Wörter in Anführungszeichen und verhindert so weitere Substitutionen.
x	Reagiert wie der Änderungswert q, trennt aber bei Leerzeichen, Tabulatorzeichen und Zeilenvorschubzeichen Wörter ab.

Wenn Sie die oben aufgelisteten Änderungswerte verwenden, bezieht sich die Änderung nur auf das erste modifizierbare Wort, sofern dem Änderungswert g der ausgewählte Änderungswert nicht vorangestellt ist.

Wird ein Protokollverweis ohne Ereignisspezifikation angegeben (zum Beispiel !\$), verwendet die Shell den vorherigen Befehl als Ereignis. Ist in derselben Zeile bereits ein Protokollverweis vorhanden, wiederholt die Shell den vorherigen Verweis. Demzufolge ergibt folgende Anweisung das erste und letzte Argument des Befehls, der mit ?foo? übereinstimmt.

```
! ?foo? ^ !$
```

Eine spezielle Abkürzung eines Protokollverweises wird verwendet, wenn es sich bei dem ersten Zeichen einer Eingabezeile, das kein Leerzeichen ist, um ein Winkelzeichen (^) handelt. Dies entspricht !:s^ und bietet somit eine einfache Kurzschreibweise für Substitutionen, die den Text der vorherigen Zeile betreffen. Der Befehl ^ lb^ lib korrigiert die Schreibweise von lib im Befehl.

Falls erforderlich, kann eine Protokollsubstitution in geschweifte Klammern ({ }) gesetzt werden, um sie von nachfolgenden Zeichen abzutrennen. Soll beispielsweise der Verweis auf den Befehl

```
ls -ld ~paul
```

verwendet werden, um den Befehl

```
ls -ld ~paula
```

auszuführen, verwenden Sie folgende Schreibweise:

```
!{1}a
```

In diesem Beispiel sucht !{1}a nach einem Befehl, der mit 1 beginnt, und fügt am Ende ein a hinzu.

*Verwendung von einfachen und doppelten Anführungszeichen:*

Sie können Zeichenfolgen in einfache und doppelte Anführungszeichen einschließen, um eine weitere Interpretation aller oder einiger Substitutionen zu verhindern.

Wenn Sie Zeichenfolgen in einfache Anführungszeichen (' ') einschließen, wird keine weitere Interpretation vorgenommen. Bei der Verwendung doppelter Anführungszeichen (" ") ist eine weitere Interpretation möglich. In beiden Fällen wird der resultierende Text (ganz oder teilweise) zu einem einzigen Wort.

### **Eingabe- und Ausgabeumleitung in der C-Shell:**

Bevor die Shell einen Befehl ausführt, sucht sie in der Befehlszeile nach Umleitungszeichen. Mit diesen Sonderzeichen wird die Shell angewiesen, die Eingabe und Ausgabe umzuleiten.

Mit den folgenden Syntaxanweisungen kann die Standardeingabe und -ausgabe eines Befehls umgeleitet werden:

Eintrag	Beschreibung
< <i>Datei</i>	Öffnet die angegebene <i>Datei</i> (bei der zuerst Variable, Befehl und Dateiname ersetzt werden) als Standardeingabe.
<< <i>Wort</i>	Liest die Shellvingabe bis zu der Zeile, die mit dem Wert der Variablen <i>Wort</i> übereinstimmt. Bei der Variablen <i>Wort</i> wird keine Variablen-, Dateinamen- oder Befehlssubstitution durchgeführt. Jede Eingabezeile wird mit der Variablen <i>Wort</i> verglichen, bevor in der Zeile Substitutionen durchgeführt werden. Sofern kein Anführungszeichen (\, ", ' oder `) in der Variablen <i>Wort</i> enthalten ist, führt die Shell die Variablen- und Befehlssubstitution in den Zwischenzeilen aus. Die Zeichen \$, \ und ` können mit dem Zeichen \ angeführt werden. Lediglich das letzte Zeilenvorschubzeichen wird gelöscht. Der daraus resultierende Text wird in eine namenlose temporäre Datei gestellt, die dem Befehl als Standardeingabe übergeben wird.
> <i>Datei</i>	Verwendet die angegebene <i>Datei</i> als Standardausgabe. Ist die <i>Datei</i> nicht vorhanden, wird sie erstellt. Ist die <i>Datei</i> vorhanden, wird sie abgeschnitten, und der ursprüngliche Inhalt geht verloren. Ist die Shellvariable <i>noclobber</i> gesetzt, darf die <i>Datei</i> nicht vorhanden sein, oder es muss sich um eine zeichenorientierte Gerätedatei handeln. Andernfalls tritt ein Fehler auf. Damit wird verhindert, dass Dateien versehentlich zerstört werden. Verwenden Sie in diesem Fall das Format mit dem Ausrufezeichen (!), um die Prüfung zu unterdrücken. <i>Datei</i> wird auf dieselbe Weise wie die Eingabedateinamen von < erweitert. Mit dem Format >& werden Standardausgabe und Standardfehlerausgabe in die angegebene <i>Datei</i> umgeleitet. Im folgenden Beispiel wird dargestellt, wie die Standardausgabe nach /dev/tty und die Standardfehlerausgabe nach /dev/null umgeleitet wird. Die runden Klammern müssen angegeben werden, damit die Standardausgabe und die Standardfehler voneinander getrennt werden.
>! <i>Datei</i>	
>& <i>Datei</i>	
>&! <i>Datei</i>	<pre>% (find / -name vi -print &gt; /dev/tty) &gt;&amp; /dev/null</pre>
> > <i>Datei</i>	Verwendet die angegebene <i>Datei</i> wie > als Standardausgabe, <i>hängt</i> die Ausgabe aber an das Ende der <i>Datei</i> an. Wenn die Shellvariable <i>noclobber</i> gesetzt ist, tritt ein Fehler auf, wenn die <i>Datei</i> nicht vorhanden ist, sofern keines der Formate mit einem Ausrufezeichen (!) angegeben wurde. Andernfalls gleicht sie der Anweisung >.
> >! <i>Datei</i>	
> >& <i>Datei</i>	
> >&! <i>Datei</i>	

Ein Befehl übernimmt die Umgebung, in der die Shell aufgerufen wurde, so wie sie von den Ein-/Ausgabeparametern und durch das Auftreten des Befehls als Befehlskette geändert wurde. Deswegen haben, anders als bei einigen vorherigen Shells, Befehle, die von einem Shell-Script ausgeführt werden, standardmäßig keinen Zugriff auf den Text der Befehle. Stattdessen empfangen sie die ursprüngliche Standardeingabe der Shell. Verwenden Sie für die Darstellung von Inline-Daten den <<-Mechanismus, damit Shellbefehlsdateien als Komponenten in Befehlsketten eingesetzt werden können und die Shell die Eingabe im Blockverfahren lesen kann. Es ist zu beachten, dass die Standardeingabe für einen Befehl, der im Hintergrund ausgeführt wird, nicht in die leere Datei /dev/null geändert wird. Stattdessen bleibt die ursprüngliche Standardeingabe der Shell die Standardeingabe.

Wenn Sie die Standardfehlerausgabe über eine Befehlskette an die Standardausgabe umleiten möchten, müssen Sie das Format |& (Pipe-Symbol, Et-Zeichen) an Stelle des Formats | (nur Pipe-Symbol) verwenden.

#### Flusssteuerung in der C-Shell:

Die Shell enthält Befehle, mit denen der Steuerungsfluss in Befehlsdateien (Shell-Scripts) und (eingeschränkt aber effektiv) der Steuerungsfluss von der Eingabe in der Shellbefehlszeile reguliert werden kann. Durch alle diese Befehle wird die Shell angewiesen, ihre Eingabe erneut zu lesen oder sie zu überspringen.

Die Anweisungen **foreach**, **switch** und **while** sowie die Form **if-then-else** der Anweisung **if** erfordern, dass die Hauptschlüsselwörter in einem einfachen Befehl in einer Eingabezeile angegeben werden.

Kann eine Shellvingabe nicht durchsucht werden, stellt die Shell die Eingabe, wenn eine Schleife gelesen wird, in den Puffer und durchsucht den internen Puffer, wenn von der Schleife aus ein erneutes Lesen erforderlich ist. In dem hier möglichen Ausmaß sind die Befehle **goto** (rückwärts) bei Eingaben, die nicht durchsucht werden können, erfolgreich.

## Sicherheit des Betriebssystems

Das Ziel der Computersicherheit ist der Schutz der Daten, die auf dem Computer gespeichert sind.

Die Datensicherheit zielt auf Folgendes ab:

Eintrag	Beschreibung
Integrität	Der Wert aller Informationen hängt von deren Genauigkeit ab. Wenn an den Daten nicht autorisierte Änderungen vorgenommen werden, verlieren diese Daten teilweise oder vollständig ihren Wert.
Vertraulichkeit	Der Wert zahlreicher Informationen hängt von deren Geheimhaltung ab.
Verfügbarkeit	Informationen müssen stets verfügbar sein.

Es ist hilfreich, Sicherheitsmaßnahmen zu planen und zu implementieren, bevor das System verwendet wird. Das Ändern von Sicherheitsrichtlinien ist sehr zeitaufwendig. Durch eine sorgfältige Planung können Sie später sehr viel Zeit einsparen.

### Identifikation und Authentifizierung

Ihre Identität wird durch Identifikation und Authentifizierung festgestellt.

Sie müssen sich am System anmelden. Sie geben Ihren Benutzernamen und Ihr Kennwort ein, wenn ein solches für Ihren Account definiert ist. In einem sicheren System sollten alle Accounts ein Kennwort haben oder andernfalls ungültig gemacht werden. Wenn Sie das richtige Kennwort eingeben, werden Sie an diesem Account angemeldet und erhalten die Zugriffsberechtigungen und Privilegien, die für den Account definiert sind.

Da das Kennwort der einzige Schutz für Ihren Account ist, sollten Sie Ihr Kennwort sorgfältig wählen und schützen. Viele Versuche, in das System einzudringen, beginnen mit dem Erraten von Kennwörtern. Das Betriebssystem verfügt über einen effizienten Kennwortschutz, indem die Benutzerkennwörter getrennt von anderen Benutzerinformationen gespeichert werden. Die verschlüsselten Kennwörter und andere sicherheitsrelevante Daten für Benutzer werden in der Datei `/etc/security/passwd` gespeichert. Auf diese Datei darf nur Root zugreifen. Mit diesem eingeschränkten Zugriff auf verschlüsselte Kennwörter ist es unbefugten Personen nicht möglich, das Kennwort mit einem Programm zu entschlüsseln, das systematisch alle möglichen und naheliegenden Kennwörter testet.

Durch wiederholte Anmeldeversuche bei einem Account ist es trotzdem möglich, Kennwörter zu erraten. Wenn Sie ein triviales Kennwort auswählen oder Ihr Kennwort nur selten ändern, können solche Versuche leicht zum Erfolg führen.

### Anmelde-IDs

Das Betriebssystem kann Benutzer auch anhand der *Anmelde-ID* identifizieren.

Mit der Anmelde-ID ist es dem System möglich, alle Benutzeraktionen bis hin zur Quelle zu verfolgen. Nach der Anmeldung am System, aber vor der Ausführung des ersten Benutzerprogramms, setzt das System die Anmelde-ID des Prozesses auf die in der Benutzerdatenbank gefundene Benutzer-ID. Alle nachfolgenden Prozesse in der Anmeldesitzung erhalten diese ID als Kennung. Anhand dieser Kennungen können alle Aktivitäten verfolgt werden, die von der Anmelde-ID durchgeführt werden.

Sie können die *aktuelle Benutzer-ID*, die *reale Benutzer-ID*, die *aktuelle Gruppen-ID*, die *reale Gruppen-ID* und die *zusätzliche Gruppen-ID* in der Sitzung zurücksetzen. Es ist jedoch nicht möglich, die Anmelde-ID zu ändern.

### Unbeaufsichtigte Terminals

Alle Systeme bieten Schwachstellen, wenn Terminals mit aktiven Anmeldungen unbeaufsichtigt bleiben. Schwerwiegende Probleme treten auf, wenn der Systemverwalter ein Terminal unbeaufsichtigt lässt, das mit Rootberechtigung aktiviert wurde. Im Allgemeinen sollten sich Benutzer jedes Mal abmelden, wenn sie ihr Terminal verlassen.



Durch die Angabe der Parameter **TMOOUT** und **TIMEOUT** in der Datei `/etc/profile` kann erzwungen werden, dass ein Terminal nach einem bestimmten Zeitraum ohne Aktivitäten selbständig eine Abmeldung durchführt. Der Parameter **TMOOUT** wird in der Korn-Shell (**ksh**) und der Parameter **TIMEOUT** in der Bourne-Shell (**bsh**) verwendet.

Im folgenden Beispiel, einem Auszug aus der Datei `.profile`, wird erzwungen, dass das Terminal nach einer Stunde ohne Aktivitäten eine Abmeldung durchführt:

```
T0=3600
echo "Setting Autologout to $T0"
TIMEOUT=$T0
TMOOUT=$T0
export TIMEOUT TMOOUT
```

**Anmerkung:** Sie können die Werte der Parameter **TMOOUT** und **TIMEOUT** in der Datei `/etc/profile` durch die Angabe anderer Werte in der Datei `.profile` im Ausgangsverzeichnis außer Kraft setzen.

#### **Zugehörige Konzepte:**

„Variablensubstitution in der Bourne-Shell“ auf Seite 284

In der Bourne-Shell können Sie Variablensubstitutionen durchführen.

#### **Zugehörige Verweise:**

„Parametersubstitution in der Korn- bzw. POSIX-Shell“ auf Seite 240

In der Korn- bzw. POSIX-Shell können Parametersubstitutionen durchgeführt werden.

## **Dateieigner und Benutzergruppen**

Der Eigner einer Datei wird zuerst über die Benutzer-ID der Person, die die Datei erstellt hat, identifiziert.

Der Dateieigner legt fest, wer die Datei lesen, in sie schreiben (ändern) oder sie ausführen darf. Mit dem Befehl **chown** kann das Eigentumsrecht geändert werden.

Jede Benutzer-ID ist durch eine eindeutige Gruppen-ID einer Gruppe zugeordnet. Der Systemverwalter erstellt die Benutzergruppen beim Einrichten des Systems. Wenn eine neue Datei erstellt wird, ordnet das Betriebssystem der Benutzer-ID, die die Datei erstellt, der Gruppe, die den Dateieigner enthält, und einer Gruppe Andere (`others`), die alle anderen Benutzer enthält, Berechtigungen zu. Der Befehl **id** zeigt Ihre Benutzer-ID (UID), Gruppen-ID (GID) und die Namen aller Gruppen an, zu denen Sie gehören.

In Dateilisten (wie z. B. den Listen, die der Befehl **ls** generiert) werden die Benutzergruppen immer in der folgenden Reihenfolge dargestellt: Benutzer, Gruppe und Andere. Wenn Sie Ihren Gruppennamen feststellen müssen, können Sie den Befehl **groups** verwenden, der alle Gruppen für eine Benutzer-ID anzeigt.

#### **Eigentumsrecht einer Datei oder eines Verzeichnisses ändern:**

Mit dem Befehl **chown** können Sie den Eigner Ihrer Dateien ändern.

Wenn Sie die Option **-R** verwenden, führt der Befehl **chown** die angeforderte Änderung vom angegebenen Verzeichnis aus rekursiv in der Verzeichnisstruktur durch. Wenn der Befehl symbolische Verbindungen findet, wird das Eigentumsrecht der Datei oder des Verzeichnisses geändert, auf das die Verbindung zeigt. Das Eigentumsrecht der symbolischen Verbindung bleibt unverändert.

**Anmerkung:** Nur Root kann den Eigner fremder Dateien ändern. Es werden keine Fehler angezeigt, wenn die Option **-f** verwendet wird.

Geben Sie beispielsweise Folgendes ein, um den Eigner der Datei `program.c` zu ändern:

```
chown jim program.c
```

Die Benutzerzugriffsberechtigungen für die Datei `program.c` gelten jetzt für `jim`. Als Eigner kann `jim` mit dem Befehl `chmod` anderen Benutzern den Zugriff auf die Datei `program.c` gewähren oder verweigern.

Die vollständige Syntax können Sie der Beschreibung des Befehls `chown` in der Veröffentlichung "Commands Reference" entnehmen.

### Zugriffsmodi für Dateien und Verzeichnisse:

Jede Datei hat einen Eigner. Wenn eine neue Datei erstellt wird, ist der Benutzer, der die Datei erstellt, der Eigner der Datei. Der Eigner legt einen *Zugriffsmodus* für die Datei fest. Mit den Zugriffsmodi können anderen Systembenutzern die Berechtigungen zum Lesen, Ändern oder Ausführen der Datei erteilt werden. Der Zugriffsmodus einer Datei kann nur vom Dateieigner oder von Benutzern mit Rootberechtigung geändert werden.

Es gibt drei Benutzergruppen: Benutzer/Eigner, Gruppe und Andere. Der Zugriff wird diesen Benutzerklassen durch eine Kombination der folgenden drei Modi erteilt: lesen (read), schreiben (write) und ausführen (execute). Beim Erstellen einer neuen Datei erhält der Benutzer, der die Datei erstellt, standardmäßig Lese-, Schreib- und Ausführungsberechtigung. Die anderen beiden Benutzergruppen erhalten Lese- und Ausführungsberechtigung. Die folgende Tabelle zeigt die Standarddateizugriffsmodi für die drei Klassen von Benutzergruppen:

Eintrag	Beschreibung		
Gruppen	Lesen	Schreiben	Ausführen
Eigner	Ja	Ja	Ja
Gruppe	Ja	Nein	Ja
Andere	Ja	Nein	Ja

Das System legt die Berechtigungen für die einzelnen Benutzer und ihre Berechtigungsstufe für jede dieser Aktivitäten fest. Zugriffsmodi werden im Betriebssystem sowohl symbolisch als auch numerisch dargestellt.

### Zugehörige Konzepte:

„Verzeichnisse“ auf Seite 505

Ein *Verzeichnis* ist eine bestimmte Art von Datei, die nur die Informationen enthält, die für den Zugriff auf Dateien und andere Verzeichnisse erforderlich sind. Folglich beansprucht ein Verzeichnis weniger Speicherplatz als andere Dateiarten.

„Dateitypen“ auf Seite 197

Die folgenden Typen von Dateien werden vom System erkannt: **reguläre Dateien**, **Verzeichnisdateien** und **Geräte-dateien**. Das Betriebssystem verwendet jedoch viele Varianten dieser grundlegenden Dateitypen.

*Symbolische Darstellung von Zugriffsmodi:*

Zugriffsmodi werden symbolisch dargestellt.

Eintrag	Beschreibung
r	Gibt den Lesezugriff an, d. h., der Benutzer kann den Inhalt einer Datei anzeigen.
w	Gibt den Schreibzugriff an, d. h., der Benutzer kann den Inhalt einer Datei ändern.
x	Gibt die Ausführungsberechtigung an. Bei ausführbaren Dateien (gewöhnliche Dateien, die Programme enthalten) bedeutet die Ausführungsberechtigung, dass das Programm ausgeführt werden kann. Bei Verzeichnissen bedeutet die Ausführungsberechtigung, dass der Inhalt des Verzeichnisses durchsucht werden kann.

Die Zugriffsmodi für Dateien und Verzeichnisse werden durch neun Zeichen dargestellt. Die ersten drei Zeichen stehen für die aktuellen Berechtigungen des **Eigners**, die zweiten drei Zeichen für die aktuellen Berechtigungen der **Gruppe** und die letzten drei Zeichen für die aktuellen Berechtigungseinstellungen **Anderer**. Ein Minuszeichen (-) in dem neunstelligen Berechtigungssatz zeigt an, dass keine Berechtigung erteilt ist. Beispielsweise gibt eine Datei mit den Zugriffsmodi `rw-r-xr-x` allen drei Gruppen Lese- und Ausführungsberechtigung und nur dem Eigner der Datei Schreibberechtigung. Dies ist die symbolische Darstellung der Standardeinstellung.

Wenn der Befehl `ls` mit dem Flag `-l` (Kleinbuchstabe L) angegeben wird, wird eine detaillierte Liste des aktuellen Verzeichnisses angezeigt. Die ersten 10 Zeichen in der mit dem Befehl `ls -l` aufgerufenen Liste zeigen den Dateityp und die Berechtigungen für jede der drei Gruppen an. Außerdem listet der Befehl `ls -l` den Eigner und die Gruppe zu jeder Datei und zu jedem Verzeichnis auf.

Das erste Zeichen gibt den Dateityp an. Die übrigen neun Zeichen enthalten Informationen zur Dateiberechtigung für alle drei Benutzergruppen. Die folgenden Symbole werden für die Darstellung des Dateityps verwendet:

Eintrag	Beschreibung
-	Reguläre Dateien
d	Verzeichnis
b	Blockorientierte Gerätedatei
c	Zeichenorientierte Gerätedateien
p	Pipe-orientierte Gerätedatei
l	Symbolische Verbindungen
s	Sockets

Das folgende Beispiel zeigt eine Liste, die der Befehl `ls -l` generieren könnte:

```
-rwxrwxr-x 2 janet acct 512 Mär 01 13:33 Januar
```

In diesem Beispiel gibt das erste Minuszeichen (-) eine reguläre Datei an. Die folgenden neun Zeichen `rwxrwxr-x` stellen die Zugriffsmodi für Benutzer, Gruppe und Andere dar. `janet` ist Dateieigner und `acct` ist der Name von Janets Gruppe. `512` ist die Dateigröße in Bytes. `Mär 01 13:33` gibt das Datum und die Uhrzeit der letzten Änderung an, und `Januar` ist der Dateiname. Die Zahl `2` gibt die Anzahl der vorhandenen Verbindungen zur Datei an.

*Numerische Darstellung von Zugriffsmodi:*

Die numerische Darstellung ist 4 für den Lesezugriff, 2 für den Schreibzugriff und 1 für die Ausführungsberechtigung. Die zwischen 1 und 7 liegende Gesamtsumme gibt den Zugriffsmodus für jede Gruppe (Benutzer, Gruppe und Andere) an.

Die folgende Tabelle enthält die numerischen Werte für die verschiedenen Zugriffsebenen:

Gesamtsumme	Lesen	Schreiben	Ausführen
0	-	-	-
1	-	-	1
2	-	2	-
3	-	2	1
4	4	-	-
5	4	-	1
6	4	2	-
7	4	2	1

Wenn eine Datei erstellt wird, erhält sie den Standardzugriffsmodus 755. Diese Angabe wird wie folgt interpretiert: Der Benutzer hat Lese-, Schreib- und Ausführungsberechtigung (4+2+1=7), die Gruppe Lese- und Ausführungsberechtigung (4+1=5), und alle anderen Benutzer haben Lese- und Ausführungsberechtigung (4+1=5). Führen Sie zum Ändern der Zugriffsberechtigungsmodi für eigene Dateien den Befehl **chmod** (Change Mode, Modus ändern) aus.

### Gruppeninformationen anzeigen:

Mit dem Befehl **lsgroup** können Sie die Attribute aller Gruppen (oder angegebener Gruppen) auf dem System anzeigen. Falls ein oder mehrere Attribute nicht gelesen werden können, listet der Befehl **lsgroup** so viele Informationen wie möglich auf.

Die Attributinformationen werden als Definitionen im Format *Attribut=Wert* angezeigt. Die einzelnen Definitionen sind jeweils durch eine Leerstelle voneinander getrennt.

1. Geben Sie zum Auflisten aller Gruppen des Systems Folgendes ein:

```
lsgroup ALL
```

Das System zeigt jede Gruppe, Gruppen-ID und alle Benutzer der Gruppe in einer Liste wie der folgenden an:

```
system 0 arne,pubs,ctw,geo,root,chucka,noer,su,dea,backup,build,janice,denise
staff 1 john,ryan,flynn,daveb,jzitt,glover,maple,ken,gordon,mbrady
bin 2 root,bin
sys 3 root,su,bin,sys
```

2. Wenn Sie bestimmte Attribute für alle Gruppen anzeigen möchten, haben Sie folgende Möglichkeiten:

- Sie können die Attribute im Format *Attribut=Wert*, durch ein Leerzeichen voneinander getrennt, auflisten. Dies ist die Standarddarstellung. Geben Sie beispielsweise Folgendes ein, um die IDs und Benutzer aller Gruppen im System aufzulisten:

```
lsgroup -a id users ALL | pg
```

Es wird eine Liste ähnlich der folgenden angezeigt:

```
system id=0 users=arne,pubs,ctw,geo,root,chucka,noer,su,dea,backup,build
staff id=1 users=john,ryan,flynn,daveb,jzitt,glover,maple,ken
```

- Sie können die Informationen auch im Zeilengruppenformat auflisten. Geben Sie beispielsweise Folgendes ein, um die IDs und die Benutzer aller Gruppen im System im Zeilengruppenformat aufzulisten:

```
lsgroup -a -f id users ALL | pg
```

Es wird eine Liste ähnlich der folgenden angezeigt:

```
system:
 id=0
 users=pubs,ctw,geo,root,chucka,noer,su,dea,backup,build
```

```
staff:
```

```
id=1
users=john,ryan,flynn,daveb,jzitt,glover,maple,ken
```

```
bin:
id=2
users=root,bin
```

```
sys:
id=3
users=root,su,bin,sys
```

3. Sie können für die Anzeige aller Attribute einer bestimmten Gruppe zwischen zwei Darstellungsarten wählen:

- Sie können die einzelnen Attribute im Format *Attribut=Wert*, durch ein Leerzeichen voneinander getrennt, auflisten. Dies ist die Standarddarstellung. Geben Sie beispielsweise Folgendes ein, um alle Attribute der Gruppe "system" aufzulisten:

```
lsgroup system
```

Es wird eine Liste ähnlich der folgenden angezeigt:

```
system id=0 users=arne,pubs,ctw,geo,root,chucka,noer,su,dea,backup,build,janice,denise
```

- Sie können die Informationen auch im Zeilengruppenformat auflisten. Geben Sie beispielsweise Folgendes ein, um alle Attribute der Gruppe bin im Zeilengruppenformat aufzulisten:

```
lsgroup -f system
```

Es wird eine Liste ähnlich der folgenden angezeigt:

```
system:
id=0 users=arne,pubs,ctw,geo,root,chucka,noer,su,dea,backup,build,janice,denise
```

4. Geben Sie Folgendes ein, um bestimmte Attribute einer bestimmten Gruppe aufzulisten:

```
lsgroup -a Attribut Gruppe
```

Geben Sie Folgendes ein, um beispielsweise die ID und die Benutzer der Gruppe bin aufzulisten:

```
lsgroup -a id users bin
```

Es wird eine Liste ähnlich der folgenden angezeigt:

```
bin id=2 users=root,bin
```

Die vollständige Syntax können Sie der Beschreibung des Befehls **lsgroup** in der Veröffentlichung "Commands Reference" entnehmen.

### Datei- und Verzeichnisberechtigungen ändern:

Mit dem Befehl **chmod** können Sie die Berechtigungen Ihrer Dateien ändern.

1. Geben Sie Folgendes ein, um den Dateien chap1 und chap2 eine Art der Zugriffsberechtigung hinzuzufügen:

```
chmod g+w chap1 chap2
```

Dieser Befehl erteilt den Mitgliedern der Gruppe Schreibberechtigung für die Dateien chap1 und chap2.

2. Geben Sie Folgendes ein, um gleichzeitig mehrere Berechtigungsänderungen für das Verzeichnis meinverz vorzunehmen:

```
chmod go-w+x meinverz
```

Dieser Befehl verweigert (-) den Gruppenmitgliedern (g) und anderen Benutzern (o) die Berechtigung zum Erstellen und Löschen von Dateien (w) im Verzeichnis meinverz und erteilt ihnen die Berechtigung (+) für das Durchsuchen und Verwenden des Verzeichnisses meinverz in einem Pfadnamen (x). Dieser Befehl entspricht der folgenden Befehlsfolge:

```
chmod g-w meinverz
chmod o-w meinverz
chmod g+x meinverz
chmod o+x meinverz
```

3. Wenn Sie beispielsweise nur dem Eigner erlauben möchten, eine Shellprozedur mit dem Namen **cmd** als Befehl auszuführen, geben Sie Folgendes ein:

```
chmod u=rwx,go= cmd
```

Dieser Befehl erteilt dem Eigner der Datei Lese-, Schreib- und Ausführungsberechtigung (**u=rwx**). Außerdem verweigert er der Gruppe und anderen Benutzern die Berechtigung für den Zugriff auf **cmd** in jeglicher Form (**go=**).

4. Wenn Sie zum Ändern der Berechtigungen für die Datei **text** die numerische Form des Befehls **chmod** verwenden möchten, geben Sie Folgendes ein:

```
chmod 644 text
```

Dieser Befehl erteilt dem Eigner Lese- und Schreibberechtigung und der Gruppe und anderen Benutzern Leseberechtigung.

Die vollständige Syntax können Sie der Beschreibung des Befehls **chmod** in der Veröffentlichung "Commands Reference" entnehmen.

## Zugriffssteuerungslisten

Die Zugriffssteuerung umfasst geschützte Informationsquellen, die angeben, wem die Zugriffsberechtigung für solche Ressourcen erteilt werden kann.

Das Betriebssystem ermöglicht die Festlegung verschiedener Sicherheitsstufen für unbedingt erforderliche und vertrauliche Daten. Der Eigner einer Informationsquelle kann anderen Benutzern Lese- und Schreibzugriffsberechtigungen für diese Ressourcen erteilen. Ein Benutzer mit Zugriffsberechtigungen für diese Ressource kann diese Berechtigungen an andere Benutzer übertragen. Diese Sicherheitsstrategie ermöglicht einen benutzergesteuerten Informationsfluss im System. Der Eigner einer Informationsquelle definiert die Zugriffsberechtigungen für das Objekt.

Benutzer können auf diese Weise nur auf eigene Objekte zugreifen. Normalerweise erhalten die Benutzer Gruppenberechtigungen oder Standardberechtigungen für eine Ressource. Die Hauptaufgabe bei der Verwaltung der Zugriffssteuerung besteht darin, die Gruppenzugehörigkeit von Benutzern zu definieren, da über die Zugehörigkeit die Zugriffsberechtigungen für fremde Dateien festgelegt werden.

### Zugriffssteuerungslisten für Dateisystemobjekte:

Dateisystemobjekte sind in der Regel einer Zugriffssteuerungsliste (ACL, Access Control List) zugeordnet, die normalerweise eine Reihe von Zugriffssteuerungseinträgen (ACE, Access Control Entry) enthält. Jeder ACE definiert eine Identität und die zugehörigen Zugriffsberechtigungen.

Zur Verwaltung von Zugriffssteuerungslisten verwenden Sie die Befehle **aclget**, **acledit**, **aclput** und **aclconvert**.

Eine ACL wird typischerweise vom physischen Dateisystem (PFS, Physical File System) auf dem Datenträger gespeichert und verwaltet. Das Betriebssystem AIX stellt eine Infrastruktur bereit, in der physische Dateisysteme mehrere ACL-Typen unterstützen und verwalten können. Das mit AIX bereitgestellte JFS2-Dateisystem unterstützt zwei Typen von ACLs:

- AIXC
- NFS4

Ältere Dateisysteme, wie sie in den früheren Releases von AIX verwendet wurden, unterstützen nur den ACL-Typ AIXC. Die ACL-Typen werden ausführlich in der Veröffentlichung *Security* beschrieben.

## ACL-Typ AIXC:

Der ACL-Typ AIXC (AIX Classic) unterstützt das in den früheren Releases von AIX definierte ACL-Verhalten. Dieser ACL-Typ umfasst die regulären Basismodusbits und erweiterte Berechtigungen (ACEs).

Mit den erweiterten Berechtigungen kann der Dateizugriff für bestimmte Benutzer oder Gruppen zugelassen oder verhindert werden, ohne die Basisberechtigungen zu ändern.

**Anmerkung:** Die AIXC-ACL für eine Datei darf die Größe einer Speicherseite (ca. 4096 Bytes) nicht überschreiten.

Der Befehl **chmod** kann bei Verwendung im numerischen Modus (Oktalangaben) die Basisberechtigungen und -attribute festlegen. Die Subroutine **chmod**, die der Befehl aufruft, inaktiviert die erweiterten Berechtigungen. Die erweiterten Berechtigungen sind inaktiviert, wenn Sie den numerischen Code des Befehls **chmod** für eine Datei verwenden, die eine ACL besitzt. Der symbolische Modus des Befehls **chmod** inaktiviert die erweiterten Berechtigungen nicht, wenn die zugehörige ACL den Typ AIXC hat. Nähere Informationen zum numerischen und symbolischen Modus finden Sie in der Beschreibung des Befehls **chmod**. Weitere Informationen zum Befehl **chmod** finden Sie unter **chmod**.

### Basisberechtigungen

Die AIXC-ACL-spezifischen Basisberechtigungen sind die traditionellen Dateizugriffsmodi, die dem Dateieigner, der Dateigruppe und anderen Benutzern zugeordnet werden. Die Zugriffsmodi sind Lesen (r), Schreiben (w) und Ausführen/Suchen (x).

**Anmerkung:** Die Basisberechtigungen des ACL-Typs AIXC entsprechen den Dateimodusbits, die in den I-Node-Headern des Dateisystemobjekts gespeichert sind. Die Informationen in den Basismodusbits sind also identisch mit dem Wert, den das Dateisystem zurückgibt, wenn Sie den Befehl **stat** für ein Dateisystemobjekt ausführen.

In Zugriffssteuerungslisten werden die Basisberechtigungen mit dem Parameter **Modus** im Format **rwX** angezeigt, wobei ein Minuszeichen (-) jede nicht erteilte Berechtigung ersetzt:

```
base permissions:
 owner(name): Modus
 group(group): Modus
 others: Modus
```

### Attribute

Einer Zugriffssteuerungsliste können drei Attribute hinzugefügt werden:

#### **setuid (SUID)**

Set-user-ID. Bit für den Definitionsmodus für Benutzer-ID. Dieses Attribut setzt die verwendeten und gespeicherten Benutzer-IDs des Prozesses während der Ausführung auf die Eigner-ID der Datei.

#### **setgid (SGID)**

Set-group-ID. Bit für den Definitionsmodus für Gruppen-ID. Dieses Attribut setzt die verwendeten und gespeicherten Gruppen-IDs des Prozesses während der Ausführung auf die Gruppen-ID der Datei.

#### **savetext (SVTX)**

Speichert den Text im Textdateiformat.

Die zuvor genannten Attribute werden im folgenden Format hinzugefügt:

```
attributes: SUID, SGID, SVTX
```

### Erweiterte Berechtigungen

Mit den erweiterten Berechtigungen einer AIXC-ACL kann der Eigner einer Datei den Zugriff auf diese Datei genauer definieren. Erweiterte Berechtigungen ändern die Basisdateiberechtigungen

(Eigner, Gruppe, Andere), indem sie für einzelne Benutzer, Gruppen oder Benutzer/Gruppe-Kombinationen Zugriffsmodi erteilen, verweigern oder angeben. Die Berechtigungen können mit Schlüsselwörtern geändert werden.

Die Schlüsselwörter `permit`, `deny` und `specify` sind wie folgt definiert:

**permit**

Erteilt dem Benutzer oder der Gruppe den angegebenen Zugriff auf die Datei.

**deny** Verwehrt dem Benutzer oder der Gruppe den angegebenen Zugriff auf die Datei.

**specify**

Definiert präzise den Dateizugriff für den Benutzer oder die Gruppe.

Wenn einem Benutzer mit dem Schlüsselwort `deny` oder `specify` ein bestimmter Zugriff verweigert wird, kann kein anderer Eintrag diese Zugriffsverweigerung außer Kraft setzen.

Zum Aktivieren der erweiterten Berechtigungen muss das Schlüsselwort "enabled" in der Zugriffssteuerungsliste (ACL) angegeben werden. Der Standardwert ist "disabled".

Die erweiterten Berechtigungen werden in einer Zugriffssteuerungsliste vom Typ AIXC in folgendem Format angegeben:

```
extended permissions:
 enabled | disabled
 permit Modus Benutzerinfo...:
 deny Modus Benutzerinfo...:
 specify Modus Benutzerinfo...:
```

Jeder `permit`-, `deny`- und `specify`-Eintrag steht in einer eigenen Zeile. Das Format des Parameters **Modus** ist `rwX`, wobei jede nicht angegebene Berechtigung durch ein Minuszeichen (-) ersetzt werden kann. Der Parameter **Benutzerinfo** wird als `u:Benutzername`, `g:Gruppenname` oder als durch Kommas getrennte Kombination von `u:Benutzername` und `g:Gruppenname` angegeben.

**Anmerkung:** Wenn in einem Eintrag mehrere Benutzernamen angegeben sind, kann dieser Eintrag nicht für eine Zugriffssteuerungsentscheidung verwendet werden, da einem Prozess nur eine Benutzer-ID zugeordnet ist.

### ACL-Typ NFS4:

Das JFS2-Dateisystem in AIX unterstützt auch den ACL-Typ NFS4. Diese ACL-Implementierung folgt der ACL-Definition, die im RFC für das Protokoll NFS4 Version 4 festgelegt ist.

Mit dieser Art von ACL haben Sie die Möglichkeit, die Zugriffsberechtigungen feiner zu steuern. Außerdem unterstützt diese ACL Funktionen wie Vererbung. Eine ACL des Typs NFS4 enthält einen Bereich von ACEs. Jeder ACE definiert die Zugriffsberechtigungen für eine Identität. Gemäß RFC setzt sich ein NFS4-ACE aus den folgenden Hauptkomponenten zusammen:

```
struct nfsace4 {
 acetype4 type;
 aceflag4 flag;
 acemask4 access_mask;
 utf8str_mixed who;
};
```

Erläuterung:

**type** Bitmaske, die den Typ des ACE definiert. Hier wird beispielsweise definiert, ob der ACE den Zugriff zulässt oder verweigert.

**flag** Bitmaske, die die Vererbungsmerkmale des ACE beschreibt. Sie definiert, ob der ACE für das Dateisystemobjekt und/oder die Kindobjekte gilt.



## access\_mask

Bitmaske, die die möglichen Zugriffsberechtigungen definiert. Zu den definierten Berechtigungen gehören read, write, execute, create, delete, create child, delete child, etc.

**who** Diese auf null endende Zeichenfolge definiert die Identität der Person, für den dieser ACE bestimmt ist. Laut RFC ist die Größe dieser Zeichenfolge nicht beschränkt. Die ungenaue Definition lässt die Definition von Domänen in NFS-Netzen der Version 4 für die Verwaltung der Zugriffssteuerung zu. In der Regel interpretiert AIX diese Zeichenfolge nicht, und jeder ACE wird einer in AIX bekannten Identität (wie z. B. **UID** oder **GID**) zugeordnet. Es wird erwartet, dass das NFS-Dateisystem der Version 4 diese Zeichenfolgen bei Bedarf interpretiert, um sie in Benutzer- bzw. Gruppen-IDs zu konvertieren, die dem Betriebssystem entsprechen. In AIX sind nur einige der Sonderzeichenfolgen für **who**, die im RFC definiert sind, bekannt.

Verwenden Sie in AIX die Befehle **aclget**, **acledit**, **aclput** und **aclconvert**, um NFS4-ACLs zu verwalten.

**Anmerkung:** Die ACL der Datei wird mit jedem Typ des Befehls **chmod** gelöscht.

## Zugriffssteuerungsliste - Beispiel für AIXC:

Im Folgenden finden Sie ein Beispiel für eine Zugriffssteuerungsliste des Typs AIXC.

Beispiel für eine ACL des Typs AIXC:

```
attributes: SUID
base permissions:
 owner(frank): rw-
 group(system): r-x
 others: ---
extended permissions:
 enabled
 permit rw- u:dhs
 deny r-- u:chas, g:system
 specify r-- u:john, g:gateway, g:mail
 permit rw- g:account, g:finance
```

Die Elemente der ACL sind im Folgenden beschrieben:

- Die erste Zeile gibt an, dass das Definitionsbit für Benutzer-ID (setuid) eingeschaltet ist.
- In der nächsten Zeile werden die Basisberechtigungen eingeführt. Die Angabe dieser Zeile ist optional.
- Die nächsten drei Zeilen enthalten die Basisberechtigungen. Die in runden Klammern angegebenen Eigner- und Gruppennamen sind ausschließlich zur Information bestimmt. Werden diese Namen geändert, führt dies nicht zu einer Änderung des Dateieigners oder der Dateigruppe. Diese Dateattribute können nur mit den Befehlen **chown** und **chgrp** geändert werden. Weitere Informationen zu diesen Befehlen finden Sie in den Beschreibungen der Befehle **chown** und **chgrp** in der Veröffentlichung "Commands Reference".
- In der nächsten Zeile werden die erweiterten Berechtigungen eingeführt. Die Angabe dieser Zeile ist optional.
- Die nächste Zeile gibt an, dass die im Folgenden aufgeführten erweiterten Berechtigungen aktiv sind.
- In den letzten vier Zeilen sind die erweiterten Einträge angegeben.
- Der erste erweiterte Eintrag erteilt dem Benutzer dhs Lese- (r) und Schreibzugriff (w) für die Datei.
- Der zweite erweiterte Eintrag verweigert dem Benutzer chas den Lesezugriff (r) nur dann, wenn er Mitglied der Gruppe "system" ist.
- Der dritte erweiterte Eintrag erteilt dem Benutzer john Lesezugriff (r), wenn er sowohl Mitglied der Gruppe gateway als auch der Gruppe mail ist. Die erweiterte Berechtigung gilt nicht, wenn der Benutzer john nicht mehr zu beiden Gruppen gehört.
- Der letzte erweiterte Eintrag erteilt allen Benutzern, die den **beiden** Gruppen account und finance angehören, Lese- (r) und Schreibberechtigung (w).

**Anmerkung:** Einem Prozess können mehrere erweiterte Einträge zugeordnet werden, wobei beschränkende Modi gegenüber berechtigenden Modi Vorrang haben.

Nähere Informationen und die vollständige Syntax finden Sie in der Beschreibung des Befehls **acledit** in der Veröffentlichung *Commands Reference, Volume 1*.

### Zugriffsberechtigung für Zugriffssteuerungslisten:

Der Eigner der Informationsquelle ist für die Verwaltung der Zugriffsberechtigungen verantwortlich. Ressourcen sind durch Berechtigungsbits geschützt, die Teil des Modus eines Objekts sind.

Für ACLs des Typs AIXC werden mit den Berechtigungsbits die Zugriffsberechtigungen definiert, die einem Objekteigner, der Objektgruppe und der Standardklasse *others* erteilt werden. Der ACL-Typ AIXC unterstützt drei verschiedene Zugriffsmodi (Lesen, Schreiben und Ausführen), die gesondert zugeteilt werden können.

Wenn sich ein Benutzer mit dem Befehl **login** oder **su** an einem Benutzeraccount anmeldet, werden die diesem Account zugeordneten Benutzer-IDs und Gruppen-IDs den Prozessen des Benutzers zugeordnet. Mit diesen IDs werden die Zugriffsberechtigungen für den Prozess festgelegt.

Für Dateien, benannte Pipes und Einheiten (Geräte-dateien) mit einer zugehörigen AIX-ACL wird der Zugriff wie folgt vergeben:

- Für jeden Zugriffssteuerungseintrag in der Zugriffssteuerungsliste findet ein Abgleich zwischen der Liste der IDs und den Prozess-IDs statt. Bei einer Übereinstimmung werden dem Prozess die für den entsprechenden Eintrag definierten Berechtigungen und Einschränkungen zugeordnet. Die logischen Verknüpfungen von Berechtigungen und Einschränkungen werden für jeden übereinstimmenden Eintrag in der Zugriffssteuerungsliste berechnet. Falls der anfordernde Prozess mit keinem Eintrag in der Zugriffssteuerungsliste übereinstimmt, werden die Berechtigungen und Einschränkungen des Standardintrags zugeordnet.
- Falls die Berechtigung für den angeforderten Zugriffsmodus erteilt wird (in der Verknüpfung der Berechtigungen enthalten) und keine Einschränkung besteht (in der Verknüpfung der Einschränkungen enthalten), wird die Zugriffsberechtigung erteilt. Andernfalls wird die Zugriffsberechtigung verweigert.

Bei einer ACL des Typs AIXC besteht eine Übereinstimmung zwischen der ID-Liste einer Zugriffssteuerungsliste und einem Prozess dann, wenn alle IDs in der Liste mit dem zugehörigen Typ der für den anfordernden Prozess gültigen ID übereinstimmen. Dies ist der Fall, wenn eine ID vom Typ **USER** mit der aktuellen Benutzer-ID des Prozesses übereinstimmt und eine ID vom Typ **GROUP** mit der aktuellen Gruppen-ID des Prozesses oder einer der zusätzlichen Gruppen-IDs übereinstimmt. Eine Zugriffssteuerungsliste mit der folgenden ID-Liste

```
USER:fred, GROUP:philosophers, GROUP:software_programmer
```

stimmt beispielsweise mit einem Prozess überein, der die aktuelle Benutzer-ID **fred** und die folgende Gruppen hat:

```
philosophers, philanthropists, software_programmer, doc_design
```

Sie würde jedoch nicht mit einem Prozess übereinstimmen, der über die aktuelle Benutzer-ID **fred** und über folgende Gruppen verfügt:

```
philosophers, iconoclasts, hardware_developer, graphic_design
```

Eine Zugriffssteuerungsliste mit der folgenden ID-Liste würde jedoch mit beiden Prozessen übereinstimmen:

```
USER:fred, GROUP:philosophers
```

Dies bedeutet, dass sich die ID-Liste in den Funktionen der Zugriffssteuerungsliste aus einer Gruppe von Bedingungen zusammensetzt, die zutreffen müssen, damit der angegebene Zugriff gewährt wird.

Die eignerdefinierte Zugriffssteuerung ermöglicht die effiziente Steuerung des Zugriffs auf Informationsquellen und einen separaten Schutz für vertrauliche Daten und Datenintegrität. Die eignerdefinierte Zugriffssteuerung ist nur so wirksam, wie sie vom Benutzer definiert wird. Alle Benutzer müssen wissen, wie Zugriffsberechtigungen erteilt und verweigert und wie diese festgelegt werden.

Bei Dateisystemobjekten mit dem ACL-Typ NFS4 basieren die Zugriffsprüfungen auf diversen ACEs, aus denen sich die ACL zusammensetzt. Dies ist in den Regeln des RFC für das Protokoll NFS Version 4 festgelegt. Der Identitätsabgleich wird auf der Basis der Benutzer-ID oder Gruppen-ID oder den in der ACE definierten Sonderzeichenfolgen für who und der Berechtigungsnachweise des Prozesses durchgeführt. Wenn eine Übereinstimmung gefunden wird, werden die angeforderten Zugriffsberechtigungen mit den im ACE definierten Zugriffsberechtigungen verglichen. Die gewährten Zugriffsberechtigungen werden übernommen, und die Vergleichsoperation fährt mit dem nächsten ACE fort. Dieser Prozess wird so lange fortgesetzt, bis das Ende der ACL erreicht ist, alle angeforderten Zugriffsberechtigungen gewährt wurden oder eine der angeforderten Zugriffsberechtigungen verweigert wurde. Die folgenden Schritte beschreiben die Zugriffsprüfung, wenn einem Dateisystemobjekt eine ACL des Typs NFS4 zugeordnet ist:

1. Für jeden Zugriffssteuerungseintrag in der Zugriffssteuerungsliste findet ein Abgleich zwischen der Liste der IDs und den Prozess-IDs statt. Die Identitätsprüfungen beinhalten die Benutzer-ID oder Gruppen-ID, die in der ACE definiert ist. Wenn die Identität mit einer **Sonderzeichenfolge** wie OWNER@ definiert ist, wird dann eine Übereinstimmung gefunden, wenn der anfordernde Prozess der Eigner der Datei ist. Falls eine Übereinstimmung gefunden wird, erhält der Prozess die für diesen Eintrag definierten Zugriffsberechtigungen. Andernfalls wird der Abgleich mit dem nächsten ACE fortgesetzt.
2. Die angeforderten Zugriffsberechtigungen werden mit den aus dem ACE abgerufenen Zugriffsberechtigungen verglichen. Falls eine der angeforderten Zugriffsberechtigungen explizit vom ACE verweigert wird, wird die Zugriffsprüfung beendet, und dem anfordernden Prozess wird der Zugriff verweigert.
3. Falls einige der angeforderten Zugriffsberechtigungen vom ACE erteilt werden, werden diese Zugriffsberechtigungen aus der Rechteliste übernommen. Anschließend wird die Vergleichsoperation mit dem nächsten ACE fortgesetzt.
4. Wenn alle angeforderten Zugriffsberechtigungen von den ACEs bestätigt werden, wird der Zugriff zugelassen.
5. Falls das Ende der ACL erreicht wird, bevor alle angeforderten Zugriffsberechtigungen aufgelöst sind, wird der Zugriff verweigert.

Abgesehen von den ACL-basierten Zugriffsprüfungen können einzelne physische Dateisysteme einen berechtigungsbasierten Zugriff auf Dateisystemobjekte erlauben. Ein Eigner kann beispielsweise unabhängig von den vorhandenen ACL-Zugriffsberechtigungen mindestens immer berechtigt sein, die ACL zu ändern. Ein Prozess mit der Benutzer-ID 0 ist ein Rootprozess. Diese Prozesse verfügen normalerweise über alle Zugriffsberechtigungen. Fordert ein Rootprozess jedoch die Ausführungsberechtigung für ein Programm an, wird die Berechtigung nur erteilt, wenn mindestens ein Benutzer Ausführungsberechtigung besitzt.

Alle Überprüfungen der Zugriffsberechtigungen für diese Objekte finden beim ersten Zugriff auf das Objekt auf der Ebene des Systemaufrufs statt. Da der Zugriff auf SVIPC-Objekte (System V Interprocess Communication) ohne Statusangabe durchgeführt wird, werden die Berechtigungen bei jedem Zugriff überprüft. Es ist jedoch möglich, dass die Prüfungen durch das physische Dateisystem beim Öffnen des Dateisystemobjekts und nicht beim Lesen oder Schreiben des Objekts durchgeführt werden. Bei Objekten mit Dateisystemnamen muss der Name des tatsächlichen Objekts aufgelöst werden können. Die Namen können entweder relativ zum Arbeitsverzeichnis des Prozesses oder absolut zum Stammverzeichnis des Prozesses aufgelöst werden. Die Namensauflösung beginnt immer mit dem Durchsuchen eines der genannten Verzeichnisse.

#### **Befehl für das Anzeigen von Zugriffssteuerungsinformationen (Befehl `aclget`):**

Der Befehl `aclget` zeigt die Zugriffssteuerungsinformationen einer Datei an. Zu den angezeigten Informationen gehören Attribute, Basisberechtigungen und erweiterte Berechtigungen.

Geben Sie beispielsweise Folgendes ein, um die Zugriffssteuerungsinformationen der Datei `status` anzuzeigen:

```
aclget status
```

Die angezeigten Zugriffssteuerungsinformationen enthalten eine Liste mit Attributen, Basisberechtigungen und erweiterten Berechtigungen.

Die vollständige Syntax des Befehls **aclget** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

### Zugehörige Konzepte:

„Zugriffssteuerungsliste - Beispiel und Beschreibung“

Im Folgenden werden Zugriffssteuerungslisten (ACL, Access Control List) anhand eines Beispiels beschrieben.

### Zugriffssteuerungsinformationen festlegen (Befehl **aclput**):

Mit dem Befehl **aclput** können Sie die Zugriffssteuerungsinformationen für eine Datei festlegen.

**Anmerkung:** Die Zugriffssteuerungsliste für eine Datei darf die Größe einer Hauptspeicherseite (ca. 4096 Byte) nicht überschreiten.

Beispiele:

Geben Sie beispielsweise Folgendes ein, um die Zugriffssteuerungsinformationen für die Statusdatei mit den in der Datei `acldefs` gespeicherten Zugriffssteuerungsinformationen festzulegen:

```
aclput -i acldefs status
```

Geben Sie beispielsweise Folgendes ein, um die Zugriffssteuerungsinformationen für die Statusdatei mit denselben Informationen festzulegen, die Sie für die Datei `plans` verwendet haben:

```
aclget plans | aclput status
```

Nähere Informationen und die vollständige Syntax finden Sie in der Beschreibung des Befehls **aclput** in der Veröffentlichung *Commands Reference, Volume 1*.

### Zugriffssteuerungsliste - Beispiel und Beschreibung:

Im Folgenden werden Zugriffssteuerungslisten (ACL, Access Control List) anhand eines Beispiels beschrieben.

Im Folgenden sehen Sie ein Beispiel für eine ACL:

```
attributes: SUID
base permissions:
 owner(frunk): rw-
 group(system): r-x
 others: ---
extended permissions:
 enabled
 permit rw- u:dhs
 deny r-- u:chas, g:system
 specify r-- u:john, g:gateway, g:mail
 permit rw- g:account, g:finance
```

Im Folgenden sind die einzelnen Elemente der Zugriffssteuerungsliste beschrieben:

- Die erste Zeile gibt an, dass das Definitionsbit für Benutzer-ID **setuid** eingeschaltet ist.
- In der nächsten Zeile werden die Basisberechtigungen eingeführt. Die Angabe dieser Zeile ist optional.

- Die nächsten drei Zeilen enthalten die Basisberechtigungen. Die in runden Klammern angegebenen Eigner- und Gruppennamen sind ausschließlich zur Information bestimmt. Werden diese Namen geändert, führt dies nicht zu einer Änderung des Dateieigners oder der Dateigruppe. Diese Dateiattribute können nur mit den Befehlen **chown** und **chgrp** geändert werden. Weitere Informationen zu diesen Befehlen finden Sie in den Beschreibungen der Befehle **chown** und **chgrp** in der Veröffentlichung "Commands Reference".
- In der nächsten Zeile werden die erweiterten Berechtigungen eingeführt. Die Angabe dieser Zeile ist optional.
- Die nächste Zeile gibt an, dass die im Folgenden aufgeführten erweiterten Berechtigungen aktiv sind.
- In den letzten vier Zeilen sind die erweiterten Einträge angegeben. Der erste erweiterte Eintrag erteilt dem Benutzer dhs Lese- (r) und Schreibzugriff (w) für die Datei.
- Der zweite erweiterte Eintrag verweigert dem Benutzer chas den Lesezugriff (r) nur dann, wenn er Mitglied der Gruppe system ist.
- Der dritte erweiterte Eintrag erteilt dem Benutzer john Lesezugriff (r), wenn er sowohl Mitglied der Gruppe gateway als auch der Gruppe mail ist. Die erweiterte Berechtigung gilt nicht, wenn der Benutzer john nicht mehr zu beiden Gruppen gehört.
- Der letzte erweiterte Eintrag erteilt allen Benutzern, die den **beiden** Gruppen account und finance angehören, Lese- (r) und Schreibberechtigung (w).

**Anmerkung:** Einem Prozess können mehrere erweiterte Einträge zugeordnet werden, wobei beschränkende Modi gegenüber berechtigenden Modi Vorrang haben.

Die vollständige Syntax können Sie der Beschreibung des Befehls **acledit** in der Veröffentlichung *Commands Reference, Volume 1* entnehmen.

#### **Zugehörige Konzepte:**

„Befehl für das Anzeigen von Zugriffssteuerungsinformationen (Befehl **aclget**)“ auf Seite 323

Der Befehl **aclget** zeigt die Zugriffssteuerungsinformationen einer Datei an. Zu den angezeigten Informationen gehören Attribute, Basisberechtigungen und erweiterte Berechtigungen.

#### **Zugehörige Tasks:**

„Zugriffssteuerungsinformationen bearbeiten (Befehl **acledit**)“

Mit dem Befehl **acledit** können Sie die Informationen zur Zugriffsteuerung einer Datei ändern. Der Befehl zeigt die aktuellen Zugriffssteuerungsdaten an und lässt die Änderung dieser Daten durch den Dateieigner zu.

#### **Zugriffssteuerungsinformationen bearbeiten (Befehl **acledit**):**

Mit dem Befehl **acledit** können Sie die Informationen zur Zugriffsteuerung einer Datei ändern. Der Befehl zeigt die aktuellen Zugriffssteuerungsdaten an und lässt die Änderung dieser Daten durch den Dateieigner zu.

Bevor die Änderungen übernommen werden, fragt der Befehl an, ob die Verarbeitung fortgesetzt werden soll. Weitere Informationen finden Sie in der Beschreibung des Befehls **acledit** in der Veröffentlichung "Commands Reference".

**Anmerkung:** Die Umgebungsvariable *EDITOR* muss mit einem vollständigen Pfadnamen angegeben werden, da andernfalls der Abbruch des Befehls **acledit** erfolgt.

Die angezeigten Zugriffssteuerungsinformationen sind für den jeweiligen ACL-Typ spezifisch und enthalten eine Liste von Attributen, Basisberechtigungen und erweiterten Berechtigungen.

Geben Sie beispielsweise Folgendes ein, um die Zugriffssteuerungsinformationen der Datei `plans` zu editieren:

```
acledit plans
```

Die vollständige Syntax des Befehls **acledit** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

### Zugehörige Konzepte:

„Zugriffssteuerungsliste - Beispiel und Beschreibung“ auf Seite 324

Im Folgenden werden Zugriffssteuerungslisten (ACL, Access Control List) anhand eines Beispiels beschrieben.

### Terminal sperren (Befehle **lock** und **xlock**)

Mit dem Befehl **lock** können Sie Ihr Terminal sperren. Der Befehl **lock** fordert das Kennwort an, liest es und fordert dann das Kennwort zur Bestätigung ein zweites Mal an.

In der Zwischenzeit wird das Terminal gesperrt und erst dann wieder freigegeben, wenn das Kennwort erneut empfangen wurde. Der Standardwert für die Zeitsperre sind 15 Minuten. Dieser Wert kann jedoch mit dem Flag *-Anzahl* geändert werden.

**Anmerkung:** Wenn Sie mit der Schnittstelle AIXwindows arbeiten, können Sie den Befehl **xlock** auf dieselbe Weise verwenden.

Geben Sie Folgendes ein, um das Terminal über die Kennwortsteuerung zu sperren:

```
lock
```

Sie werden zweimal zur Eingabe des Kennworts aufgefordert, damit das System das Kennwort prüfen kann. Wenn Sie das Kennwort nicht innerhalb von 15 Minuten erneut eingeben, wird der Befehl mit einer Zeitlimitüberschreitung abgebrochen.

Geben Sie Folgendes ein, um das Terminal über die Kennwortsteuerung und mit einem Zeitlimit von 10 Minuten zu sperren:

```
lock -10
```

Die vollständige Syntax können Sie der Beschreibung der Befehle **lock** und **xlock** in der Veröffentlichung *Commands Reference* entnehmen.

### Befehlsübersicht für Datei- und Systemsicherheit

Im Folgenden finden Sie eine Übersicht über die Befehle für Dateisysteme und Sicherheit.

Eintrag	Beschreibung
<b>acledit</b>	Editiert die Informationen für die Zugriffssteuerung einer Datei.
<b>aclget</b>	Zeigt die Informationen für die Zugriffssteuerung einer Datei an.
<b>aclput</b>	Setzt die Informationen für die Zugriffssteuerung einer Datei.
<b>chmod</b>	Ändert die Berechtigungsmodi.
<b>chown</b>	Ändert den Benutzer, der einer Datei zugeordnet ist.
<b>lock</b>	Sperrt ein Terminal.
<b>lsgroup</b>	Zeigt die Attribute von Gruppen an.
<b>xlock</b>	Sperrt die lokale X-Station so lange, bis ein Kennwort eingegeben wird.

### Benutzerumgebung

Jeder Anmeldenname besitzt eine eigene Systemumgebung.

Die Systemumgebung ist ein Bereich, in dem Informationen gespeichert sind, die für alle Prozesse, die in einer Sitzung ausgeführt werden, gleichermaßen gelten. Es stehen verschiedene Befehle zur Verfügung, mit denen Sie Informationen zu Ihrem System anzeigen können.

## Benutzerumgebungsdateien und Anpassungsprozeduren

Die folgenden Dateien und Prozeduren helfen Ihnen bei der Anpassung der Systemumgebung.

### Systemstartdateien

Eintrag	Beschreibung
<code>/etc/profile</code>	Systemdatei mit Befehlen, die das System ausführt, wenn Sie sich anmelden.
<code>/etc/environment</code>	Systemdatei mit Variablen, die die Basisumgebung für alle Prozesse definieren.
<code>\$HOME/.profile</code>	In Ihrem Ausgangsverzeichnis gespeicherte Datei mit Befehlen, die die Systemdatei <code>/etc/profile</code> überschreibt, wenn Sie sich anmelden. Weitere Informationen finden Sie im Abschnitt "Datei <code>.profile</code> ".
<code>\$HOME/.env</code>	In Ihrem Ausgangsverzeichnis gespeicherte Datei, die die Systemdatei <code>/etc/environment</code> überschreibt und Variablen enthält, die die Basisumgebung für alle Prozesse definieren. Weitere Informationen finden Sie im Abschnitt "Datei <code>.env</code> ".

### Startdateien für AIXwindows

Eintrag	Beschreibung
<code>\$HOME/.xinitrc</code>	In Ihrem Ausgangsverzeichnis gespeicherte Datei, die die Fenster und Anwendung steuert, die beim Starten von AIXwindows gestartet werden. Weitere Informationen finden Sie im Abschnitt "Datei <code>.xinitrc</code> ".
<code>\$HOME/.Xdefaults</code>	In Ihrem Ausgangsverzeichnis gespeicherte Datei, die das Erscheinungsbild und die Funktionsweise der AIXwindows-Ressourcen steuert. Nähere Informationen hierzu finden Sie im Abschnitt „Datei <code>.Xdefaults</code> “ auf Seite 337".
<code>\$HOME/.mwmrc</code>	In Ihrem Ausgangsverzeichnis gespeicherte Datei, die Tastenbelegungen, Maustastenbelegungen und Menüdefinitionen für den Fenstermanager definiert. Nähere Informationen hierzu finden Sie im Abschnitt „Datei <code>.mwmrc</code> “ auf Seite 338.

### Anpassungsprozeduren

Eintrag	Beschreibung
<code>PS1</code>	Normale Eingabeaufforderung
<code>PS2</code>	Aufforderung für weitere Eingaben
<code>PS3</code>	Rooteingabeaufforderung
<code>chfont</code>	Ändert die von der Anzeige verwendete Schriftart beim Neustart des Systems.
<code>stty</code>	Setzt die Betriebsparameter der Workstation, setzt sie zurück und listet sie auf.

### Systemeinheiten auflisten (Befehl `lscfg`)

Mit dem Befehl `lscfg` können Sie Namen, Position und Beschreibung der in der aktuellen Konfiguration vorhandenen Einheiten anzeigen. Das Sortierkriterium für die Liste ist die Einheitenposition.

Geben Sie beispielsweise Folgendes an der Eingabeaufforderung ein, um die im System konfigurierten Einheiten aufzulisten:

```
lscfg
```

Das System gibt Informationen wie die folgenden aus:

```
INSTALLED RESOURCE LIST
```

```
The following resources are installed on your machine.
```

```
+/- = Added/Deleted from Diagnostic Test List.
```

```
* = NOT Supported by Diagnostics.
```

```
Model Architecture: chrp
```

```
Model Implementation: Multiple Processor, PCI bus
```

```
+ sysplanar0 00-00 CPU Planar
+ fpa0 00-00 Floating Point Processor
+ mem0 00-0A Memory Card
+ mem1 00-0B Memory Card
```

```

+ ioplanar0 00-00 I/O Planar
+ rs2320 00-01 RS232 Card
+ tty0 00-01-0-01 RS232 Card Port
- tty1 00-01-0-02 RS232 Card Port
..
..
..

```

Die Einheitenposition ist nicht das einzige Sortierkriterium. Die Liste ist auch nach Eltern/Kind-Hierarchie sortiert. Falls die Elterneinheit mehrere Kindeinheiten besitzt, sind die Kindeinheiten nach Einheitenposition sortiert. Wenn die Kindeinheiten dieselbe Einheitenposition haben, werden sie in der Reihenfolge aufgeführt, in der sie von der Software abgerufen wurden. Mit dem Flag **-l** können Sie Informationen zu einer bestimmten Einheit anzeigen. Geben Sie beispielsweise Folgendes an der Eingabeaufforderung ein, um die Informationen für die Einheit **sysplanar0** anzuzeigen:

```
lscfg -l sysplanar0
```

Das System gibt Informationen wie die folgenden aus:

```

DEVICE LOCATION DESCRIPTION
sysplanar0 00-00 CPU Planar

```

Mit dem Befehl **lscfg** können auch elementare Produktdaten (VPD, Vital Product Data) wie Teilenummern, Seriennummern und EC-Stufen (Engineering Change, technische Änderung) angezeigt werden. Bei einigen Einheiten werden die elementaren Produktdaten automatisch erfasst und der Systemkonfiguration hinzugefügt. Bei anderen Einheiten werden die elementaren Produktdaten manuell eingegeben. Die Angabe ME vor den Daten zeigt an, dass die Daten manuell eingegeben wurden.

Geben Sie beispielsweise Folgendes an der Eingabeaufforderung ein, um die elementaren Produktdaten für die im System konfigurierten Einheiten aufzulisten:

```
lscfg -v
```

Das System gibt Informationen wie die folgenden aus:

```

INSTALLED RESOURCE LIST WITH VPD
The following resources are installed in your machine.
 Model Architecture: chrp
 Model Implementation: Multiple Processor, PCI bus
sysplanar0 00-00 CPU Planar
 Part Number.....342522
 EC Level.....254921
 Serial Number.....353535
fpa0 00-00 Floating Point Processor
mem0 00-0A Memory Card
 EC Level.....990221
.
.
.

```

Die vollständige Syntax des Befehls **lscfg** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

## Konsolnamen anzeigen

Mit dem Befehl **lscons** können Sie den Namen der aktuellen Konsoleinheit in die Standardausgabe (in der Regel Ihr Bildschirm) schreiben.

Geben Sie an der Eingabeaufforderung beispielsweise Folgendes ein:

```
lscons
```



Das System gibt Informationen wie die folgenden aus:

```
/dev/lft0
```

Die vollständige Syntax können Sie der Beschreibung des Befehls **lscons** in der Veröffentlichung "Commands Reference" entnehmen.

### Terminalnamen anzeigen (Befehl tty)

Mit dem Befehl **tty** können Sie den Namen Ihres Terminals anzeigen.

Geben Sie an der Eingabeaufforderung beispielsweise Folgendes ein:

```
tty
```

Das System zeigt Informationen ähnlich den folgenden an:

```
/dev/tty06
```

In diesem Beispiel ist **tty06** der Name des Terminals und **/dev/tty06** die Einheitendatei, die die Schnittstelle zu diesem Terminal enthält.

Die vollständige Syntax können Sie der Beschreibung des Befehls **tty** in der Veröffentlichung *Commands Reference, Volume 5* entnehmen.

### Verfügbare Bildschirme auflisten (Befehl lsdisp)

Mit dem Befehl **lsdisp** können Sie die derzeit im System verfügbaren Bildschirme mit Bildschirm-ID, Steckplatznummer, Bildschirmnamen und Beschreibung auflisten.

Geben Sie beispielsweise Folgendes ein, um alle verfügbaren Bildschirme aufzulisten:

```
lsdisp
```

Es erscheint eine Ausgabe wie die folgende. Die Liste ist in aufsteigender Reihenfolge nach Steckplatznummer sortiert.

Name	Slot	Name	Description
ppr0	00-01	POWER_G4	Midrange Graphics Adapter
gda0	00-03	colorgda	Color Graphics Display Adapter
ppr1	00-04	POWER_Gt3	Midrange Entry Graphics Adapter

Die vollständige Syntax des Befehls **lsdisp** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

### Verfügbare Schriftarten auflisten (Befehl lsfont)

Mit dem Befehl **lsfont** können Sie eine Liste der für Ihren Bildschirm verfügbaren Schriftarten anzeigen.

Geben Sie beispielsweise Folgendes ein, um alle für den Bildschirm verfügbaren Schriftarten aufzulisten:

```
lsfont
```

Die folgende Beispielausgabe enthält die Schriftart-ID, den Dateinamen, die Glyph-Größe und die Schriftartcodierung:

FONT ID	FILE NAME	GLYPH SIZE	FONT ENCODING
====	=====	=====	=====
0	Erg22.iso1.snf	12x30	IS08859-1
1	Erg11.iso1.snf	8x15	IS08859-1

Die vollständige Syntax des Befehls **lsfont** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

## Aktuelle Softwaretastaturtabelle auflisten (Befehl `lskbd`)

Mit dem Befehl `lskbd` können Sie den absoluten Pfadnamen der derzeit im System geladenen Softwaretastaturtabelle anzeigen.

Geben Sie beispielsweise Folgendes ein, um die derzeit verwendete Tastaturtabelle anzuzeigen:

```
lskbd
```

Das folgende Beispiel zeigt eine Liste, die vom Befehl `lskbd` ausgegeben wird:

```
The current software keyboard map = /usr/lib/nls/loc/C.1ftkeymap
```

## Verfügbare Softwareprodukte auflisten (Befehl `lslpp`)

Mit dem Befehl `lslpp` können Sie Informationen zu den auf Ihrem System verfügbaren Softwareprodukten anzeigen.

Geben Sie beispielsweise an der Eingabeaufforderung Folgendes ein, um alle auf dem System verfügbaren Softwareprodukte aufzulisten:

```
lslpp -l -a
```

Es erscheint eine Ausgabe wie die folgende:

Fileset	Level	State	Description
Path: /usr/lib/objrepos			
X11_3d.gl.dev.obj		APPLIED	AIXwindows/3D GL Development Utilities
Fonts			
X11fnt.oldX.fnt		APPLIED	AIXwindows Miscellaneous X Fonts
X11mEn_US.msg		APPLIED	AIXwindows NL Message files
.			
.			
.			

Falls die Liste sehr umfangreich ist, verschiebt sich der obere Teil unter Umständen aus dem Sichtbereich. Wenn Sie die Liste seitenweise anzeigen möchten, müssen Sie den Befehl `lslpp` durch ein Pipe-Symbol mit dem Befehl `pg` verketteten. Geben Sie an der Eingabeaufforderung Folgendes ein:

```
lslpp -l -a | pg
```

Die vollständige Syntax des Befehls `lslpp` ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

## Liste der Steuertastenbelegungen für Terminal (Befehl `stty`)

Mit dem Befehl `stty` können Sie Ihre Terminaleinstellungen anzeigen. Achten Sie insbesondere auf die Tasten, die das Terminal als Steuertasten verwendet.

Geben Sie an der Eingabeaufforderung beispielsweise Folgendes ein:

```
stty -a
```

Das System zeigt Informationen ähnlich den folgenden an:

```
.
. .
intr = ^C; quit = ^\; erase = ^H; kill = ^U; eof = ^D;
eol = ^@ start = ^Q; stop = ^S; susp = ^Z; dsusp = ^Y;
reprint = ^R discard = ^O; werase = ^W; lnext = ^V
```

.  
.
.  
.

In diesem Beispiel enthalten Zeilen wie `intr = ^C`; `quit = ^\`; `erase = ^H`; die Einstellungen Ihrer Steuertasten. `^H` steht für die Rücktaste und ist so eingestellt, dass Sie Zeichen löschen können.

Falls die Liste sehr umfangreich ist, verschiebt sich der obere Teil unter Umständen aus dem Sichtbereich. Wenn Sie die Liste seitenweise anzeigen möchten, müssen Sie den Befehl `stty` durch ein Pipe-Symbol mit dem Befehl `pg` verketteten. Geben Sie an der Eingabeaufforderung Folgendes ein:

```
stty -a | pg
```

Die vollständige Syntax des Befehls `stty` ist in der Veröffentlichung *Commands Reference, Volume 5* beschrieben.

### Zugehörige Konzepte:

„Vordergrundprozesse abbrechen“ auf Seite 149

Wenn Sie einen Vordergrundprozess starten und sich danach entscheiden, den Prozess nicht zu Ende auszuführen, können Sie ihn durch Drücken der Unterbrechungstaste abbrechen. Hierbei handelt es sich gewöhnlich um die Tastenkombination Strg-C oder Strg-Rücktaste.

### Umgebungsvariablen auflisten (Befehl env)

Mit dem Befehl `env` können Sie die aktuellen Umgebungsvariablen anzeigen. Umgebungsvariablen, auf die alle Prozesse zugreifen können, werden *globale Variablen* genannt.

Geben Sie beispielsweise Folgendes ein, um alle Umgebungsvariablen mit Werten aufzulisten:

```
env
```

Es erscheint eine Ausgabe wie die folgende:

```
TMPDIR=/usr/tmp
myid=denise
LANG=en_US
UNAME=barnard
PAGER=/bin/pg
VISUAL=vi
PATH=/usr/ucb:/usr/lpp/X11/bin:/bin:/usr/bin:/etc:/u/denise:/u/denise/bin:/u/bin1
MAILPATH=/usr/mail/denise?denise has mail !!!
MAILRECORD=/u/denise/.Outmail
EXINIT=set beautify noflash nomesg report=1 showmode showmatch
EDITOR=vi
PSCH=>
HISTFILE=/u/denise/.history
LOGNAME=denise
MAIL=/usr/mail/denise
PS1=denise@barnard:${PWD}>
PS3=#
PS2=>
epath=/usr/bin
USER=denise
SHELL=/bin/ksh
HISTSIZE=500
HOME=/u/denise
FCEDIT=vi
TERM=1ft
MAILMSG=**YOU HAVE NEW MAIL. USE THE mail COMMAND TO SEE YOUR PWD=/u/denise
ENV=/u/denise/.env
```

Falls die Liste sehr umfangreich ist, verschiebt sich der obere Teil unter Umständen aus dem Sichtbereich. Wenn Sie die Liste seitenweise anzeigen möchten, müssen Sie den Befehl `env` durch ein Pipe-Symbol mit dem Befehl `pg` verketteten. Geben Sie an der Eingabeaufforderung Folgendes ein:

Die vollständige Syntax des Befehls **env** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

### Wert einer Umgebungsvariablen anzeigen (Befehl **printenv**)

Mit dem Befehl **printenv** können Sie die Werte von Umgebungsvariablen anzeigen.

Wenn Sie den Parameter **Name** verwenden, gibt das System nur den Wert der angeforderten Variablen aus. Wenn Sie den Parameter **Name** nicht verwenden, zeigt der Befehl **printenv** alle aktuellen Umgebungsvariablen an, wobei jedes **Name=Wert**-Paar in einer eigenen Zeile dargestellt wird.

Geben Sie beispielsweise Folgendes ein, um die aktuelle Einstellung der Umgebungsvariablen **MAILMSG** zu ermitteln:

```
printenv MAILMSG
```

Daraufhin gibt der Befehl den Wert der Umgebungsvariablen **MAILMSG** zurück. Beispiel:

```
SIE HABEN NEUE MAIL
```

Die vollständige Syntax des Befehls **printenv** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

### Bidirektionale Sprachen (Befehl **aixterm**)

Der Befehl **aixterm** unterstützt die bidirektionalen Sprachen Arabisch und Hebräisch.

Bidirektionale Sprachen können in beide Richtungen gelesen und geschrieben werden (von links nach rechts und von rechts nach links). Wenn Sie mit Anwendungen in arabischer oder hebräischer Sprache arbeiten möchten, müssen Sie ein Fenster öffnen und für dieses Fenster eine arabische bzw. hebräische Locale einstellen.

Die vollständige Syntax des Befehls **aixterm** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

### Befehlsübersicht für Benutzerumgebung und Systeminformationen

Im Folgenden werden Befehle für die Benutzerumgebung und Systeminformationen aufgelistet.

Eintrag	Beschreibung
<b>aixterm</b>	Ermöglicht das Arbeiten mit bidirektionalen Sprachen.
<b>env</b>	Zeigt die aktuelle Umgebung an oder legt die Umgebung für die Ausführung eines Befehls fest.
<b>lscfg</b>	Zeigt Diagnoseinformationen zu einer Einheit an.
<b>lscons</b>	Zeigt den Namen der aktuellen Konsole an.
<b>lsdisp</b>	Listet die derzeit im System verfügbaren Bildschirme auf.
<b>lsfont</b>	Listet die für den Bildschirm verfügbaren Schriftarten auf.
<b>lskbd</b>	Listet die derzeit im System geladene Tastaturliste auf.
<b>lslpp</b>	Listet Softwareprodukte auf.
<b>printenv</b>	Zeigt die Werte von Umgebungsvariablen an.
<b>stty</b>	Zeigt Systemeinstellungen an.
<b>tty</b>	Zeigt den vollständigen Pfadnamen des Terminals an.

### Anpassung der Benutzerumgebung

Das Betriebssystem stellt diverse Befehle und Initialisierungsdateien bereit, mit denen Sie das Verhalten und die Darstellung Ihrer Benutzerumgebung anpassen können.

Außerdem können Sie einige der Standardressourcen von Anwendungen anpassen, die Sie auf dem System verwenden. Die Standardeinstellungen werden beim Programmstart festgelegt. Wenn Sie die Standardwerte ändern, müssen Sie das Programm beenden und erneut starten, damit die neuen Standardwerte übernommen werden.

Nähere Informationen zum Anpassen des Verhaltens und der Darstellung von Common Desktop Environment finden Sie in der Veröffentlichung *Common Desktop Environment 1.0: Advanced User's and System Administrator's Guide*.

### **Systemstartdateien:**

Bei der Anmeldung am System definiert die Shell nach dem Lesen der installierten Initialisierungsdateien die entsprechende Benutzerumgebung. Die Merkmale der jeweiligen Benutzerumgebung werden durch die in den Umgebungsvariablen angegebenen Werte definiert. Diese Umgebung bleibt bis zum Abmelden vom System bestehen.

Die Shell verwendet beim Anmelden am Betriebssystem zwei Arten von Profildateien. Sie wertet die in den Dateien enthaltenen Befehle aus und führt sie anschließend aus, um die entsprechende Systemumgebung einzurichten. Die Dateien haben eine ähnliche Funktion, mit der Ausnahme, dass die Datei `/etc/profile` die Profilvariablen für alle Benutzer eines Systems steuert, während die Datei `.profile` die Anpassung der eigenen Umgebung zulässt.

Die Shell führt zuerst die in der Datei `/etc/environment` enthaltenen Befehle für die Konfiguration der Systemumgebung aus und wertet anschließend die Befehle aus, die in der Datei `/etc/profile` enthalten sind. Nach Ausführung dieser Dateien prüft das System, ob sich eine Datei `.profile` in Ihrem Ausgangsverzeichnis befindet. Wenn die Datei `.profile` vorhanden ist, führt das System sie aus. Die Datei `.profile` gibt an, ob auch eine Umgebungsdatei vorhanden ist. Wenn eine Umgebungsdatei vorhanden ist (die in der Regel die Dateierweiterung `.env` hat), führt das System diese Datei aus und setzt die Umgebungsvariablen.

Die Dateien `/etc/environment`, `/etc/profile` und `.profile` werden während der Anmeldung einmal ausgeführt. Die Datei `.env` hingegen wird jedes Mal ausgeführt, wenn eine neue Shell oder ein Fenster geöffnet wird.

#### *Datei /etc/environment:*

Die erste Datei, die das Betriebssystem beim Anmelden verwendet, ist die Datei `/etc/environment`. Die Datei `/etc/environment` enthält Variablen, die die Basisumgebung für alle Prozesse definieren.

Wenn ein neuer Prozess gestartet wird, stellt die Subroutine `exec` ein Array von Zeichenfolgen bereit, die das Format *Name=Wert* haben. Dieses Array von Zeichenfolgen wird als *Umgebung* bezeichnet. Jeder Name, der durch eine der Zeichenfolgen definiert wird, ist eine *Umgebungsvariable* oder *Shellvariable*. Mit der Subroutine `exec` kann die gesamte Umgebung auf einmal definiert werden.

Wenn Sie sich anmelden, setzt das System die in der Datei `/etc/environment` definierten Umgebungsvariablen, bevor es Ihr Anmeldeprofil liest, das die Dateierweiterung `.profile` hat. Die Basisumgebung umfasst die folgenden Variablen:

Eintrag	Beschreibung
<i>HOME</i>	Der vollständige Pfadname des Anmeldeverzeichnisses oder Ausgangsverzeichnisses des Benutzers. Das Anmeldeprogramm <i>login</i> setzt diese Variable auf den in der Datei <i>/etc/passwd</i> angegebenen Namen.
<i>LANG</i>	Der Name der gegenwärtig aktiven Locale. Die Variable <i>LANG</i> wird erstmalig während der Installation in der Datei <i>/etc/profile</i> gesetzt.
<i>NLSPATH</i>	Der vollständige Pfadname für Nachrichtenkataloge.
<i>LOCPATH</i>	Der vollständige Pfadname für die Tabellen für die Unterstützung in der Landessprache.
<i>PATH</i>	Die Reihenfolge der Verzeichnisse, in denen Befehle, wie z. B. <b>sh</b> , <b>time</b> , <b>nice</b> und <b>nohup</b> , nach Befehlen suchen, deren Pfadname unvollständig angegeben wurde.
<i>TZ</i>	Informationen zur Zeitzone. Die Umgebungsvariable <i>TZ</i> wird erstmalig von der Datei <i>/etc/profile</i> , dem Systemanmeldeprofil, gesetzt.

Ausführliche Informationen zur Datei */etc/environment* finden Sie in der Veröffentlichung *Files Reference*.

Datei */etc/profile*:

Die zweite Datei, die das Betriebssystem beim Anmelden verwendet, ist die Datei */etc/profile*.

Die Datei */etc/profile* steuert systemweite Standardvariablen wie die folgenden:

- Exportvariablen
- Dateierstellungsmaske (*umask*)
- Terminaltypen
- Mail-Nachrichten, die anzeigen, dass neue Mail empfangen wurde.

Die Datei */etc/profile* wird vom Systemadministrator für alle Benutzer auf dem System konfiguriert. Ausschließlich der Systemadministrator ist berechtigt, die Datei zu ändern.

Das folgende Beispiel zeigt eine typische Datei */etc/profile*:

```
#Maske für Dateierstellung definieren
umask 022
#Benachrichtigen, wenn neue Mail ankommt
MAIL=/usr/mail/$LOGNAME
#Verzeichnis /bin zur Suchfolge für die Shell hinzufügen
PATH=/usr/bin:/usr/sbin:/etc::
#Terminaltyp festlegen
TERM=1ft
#Einige Umgebungsvariablen als globale Variablen definieren
export MAIL PATH TERM
```

Ausführliche Informationen zur Datei */etc/profile* finden Sie in der Veröffentlichung *Files Reference*.

Datei *.profile*:

Die Datei *.profile* befindet sich im Ausgangsverzeichnis (*\$HOME*). Mit dieser Datei können Sie Ihre eigene Arbeitsumgebung nach Bedarf anpassen.

Da die Datei *.profile* verdeckt gespeichert ist, müssen Sie zum Auflisten der Datei den Befehl **ls -a** verwenden.

Nachdem das Programm **login** der Umgebung die Variablen *LOGNAME* (Anmeldename) und *HOME* (Anmeldeverzeichnis) hinzugefügt hat, werden die Befehle in der Datei *\$HOME/.profile* ausgeführt, sofern die Datei vorhanden ist. Die Datei *.profile* enthält Ihr Profil, das die in der Datei */etc/profile* gesetzten Variablen außer Kraft setzt. Die Datei *.profile* wird häufig verwendet, um exportierte Umge-

Umgebungsvariablen und Terminalmodi zu setzen. Sie können Ihre Umgebung anpassen. Editieren Sie dazu die Datei `.profile`. Sie können mit der Datei `.profile` die folgenden Standardeinstellungen steuern:

- Zu öffnende Shells
- Eingabeaufforderung
- Tastatursignale

Das folgende Beispiel zeigt eine typische Datei `.profile`:

```
PATH=/usr/bin:/etc:/home/bin1:/usr/lpp/tps4.0/user::
epath=/home/gsc/e3:
export PATH epath
csh
```

In diesem Beispiel werden zwei Pfadvariablen definiert (`PATH` und `epath`) und exportiert sowie eine C-Shell geöffnet (`csh`).

Außerdem kann die Datei `.profile` (oder falls diese nicht vorhanden ist, die Datei `/etc/profile`) zur Festlegung von Variablen für die Anmeldeshell verwendet werden. Andere Shellumgebungen können ebenfalls angepasst werden. Mit den Dateien `.cshrc` und `.kshrc` können respektive die C- und die Korn-Shell angepasst werden.

*Datei .env:*

Eine vierte Datei, die das Betriebssystem beim Anmelden verwendet, ist die Datei `.env`, falls in der Datei `.profile` folgende Zeile angegeben ist: `export ENV=$HOME/.env`.

Mit der Datei `.env` können Sie die Variablen für Ihre Arbeitsumgebung individuell anpassen. Da die Datei `.env` verdeckt gespeichert ist, müssen Sie zum Auflisten der Datei den Befehl `ls -a` verwenden. Weitere Informationen finden Sie in der Beschreibung des Befehls `ls` in der Veröffentlichung "Commands Reference". Die Datei `.env` enthält die einzelnen Umgebungsvariablen des Benutzers, die die in der Datei `/etc/environment` gesetzten Variablen außer Kraft setzen. Sie können die Umgebungsvariablen anpassen. Editieren Sie dazu die Datei `.env`.

Das folgende Beispiel zeigt eine typische Datei `.env`:

```
export myid=`id | sed -n -e 's/).*$/' -e 's/^.*/p'`
#set prompt: login & system name & path
if [$myid = root]
 then typeset -x PSCH='#:\${PWD}> '
 PS1="#:\${PWD}> "
 else typeset -x PSCH='>'
 PS1="$LOGNAME@\$UNAME:\${PWD}> "
 PS2=">"
 PS3="#?"
fi
export PS1 PS2 PS3
#setup my command aliases
alias ls="/bin/ls -CF" \
 d="/bin/ls -Fal | pg" \
 rm="/bin/rm -i" \
 up="cd .."
```

**Anmerkung:** Wenn Sie die Datei `.env` ändern, müssen Sie sicherstellen, dass neu erstellte Umgebungsvariablen nicht mit Standardvariablen wie `MAIL`, `PS1`, `PS2` oder `IFS` in Konflikt geraten.

### AIXwindows-Startdateien:

Unterschiedliche Computersysteme haben unterschiedliche Methoden zum Starten von X-Servern und AIXwindows.

Da unterschiedliche Computersysteme unterschiedliche Methoden zum Starten von X-Servern und AIXwindows einsetzen, muss zunächst mit dem Systemadministrator geklärt werden, wie das System gestartet wird. Gewöhnlich werden der X-Server und AIXwindows von einem Shell-Script gestartet, das automatisch ausgeführt wird, wenn die Anmeldung am System erfolgt. Es kann jedoch sein, dass der X-Server oder AIXwindows oder beide manuell gestartet werden müssen.

Wird nach dem Anmelden festgestellt, dass die Anzeige als Einzelterminal eingerichtet ist und keine Fenster angezeigt werden, kann der X-Server durch Eingabe des folgenden Befehls gestartet werden:

```
xinit
```

**Anmerkung:** Vor der Eingabe dieses Befehls muss sichergestellt werden, dass sich der Zeiger in einem Fenster befindet, das eine Systemeingabeaufforderung enthält.

Falls der X-Server nach Eingabe dieses Befehls nicht gestartet wird, wenden Sie sich an Ihren Systemadministrator, um festzustellen, ob der angegebene Suchpfad das X11-Verzeichnis mit den ausführbaren Programmen enthält. Der Pfad ist von System zu System verschieden.

Wird nach dem Anmelden festgestellt, dass ein oder mehrere Fenster ohne Rahmen angezeigt werden, kann der AIXwindows-Fenstermanager (Window Manager) durch Eingabe des folgenden Befehls gestartet werden:

```
mwm &
```

Da AIXwindows von Programmierern, die AIXwindows-Anwendungen schreiben, und Benutzern angepasst werden kann, arbeiten manche Maustasten und Funktionen möglicherweise anders als in dieser Dokumentation beschrieben. Sie können Ihre AIXwindows-Umgebung mit der folgenden Tastenkombination auf die Standardeinstellungen zurücksetzen:

Alt-Strg-Umschalttaste-!

Durch erneutes Drücken dieser Tastenkombination kann die angepasste Umgebung wieder aktiviert werden. Falls Ihr System diese Tastenkombination nicht unterstützt, können Sie die Standardeinstellungen auch mit dem Standardbasismenü wiederherstellen.

*Datei .xinitrc:*

Der Befehl **xinit** verwendet ein anpassbares Shell-Script, in dem die zu startenden X-Clientprogramme aufgelistet sind. Die Datei `.xinitrc` in Ihrem Ausgangsverzeichnis steuert die Fenster und Anwendungen, die beim Starten von AIXwindows gestartet werden.

Der Befehl **xinit** arbeitet die Shell-Scripts in der folgenden Reihenfolge ab:

1. Der Befehl **xinit** sucht zuerst nach der Umgebungsvariablen `$XINITRC`, um AIXwindows zu starten.
2. Falls die Umgebungsvariable `$XINITRC` nicht gefunden wird, sucht der Befehl **xinit** nach dem Shell-Script `$HOME/.xinitrc`.
3. Falls das Shell-Script `$HOME/.xinitrc` nicht gefunden wird, startet der Befehl **xinit** das Shell-Script `/usr/lib/X11/$LANG/xinitrc`.
4. Falls `/usr/lib/X11/$LANG/xinitrc` nicht gefunden wird, sucht der Befehl nach dem Shell-Script `/usr/lpp/X11/defaults/$LANG/xinitrc`. Sollte dieses Shell-Script auch nicht vorhanden sein, sucht der Befehl nach dem Shell-Script `/usr/lpp/X11/defaults/xinitrc`.
5. Das Shell-Script `xinitrc` startet Befehle, wie z. B. die Befehle **mwm** (AIXwindows Window Manager), **aixterm** und **xclock**.

Der Befehl **xinit** führt folgende Operationen durch:

- Er startet einen X-Server in der aktuellen Anzeige.
- Er setzt die Umgebungsvariable `$DISPLAY`.



- Er führt die Datei `xinitrc` aus, um die X-Clientprogramme zu starten.

Das folgende Beispiel zeigt den Teil der Datei `xinitrc`, den Sie anpassen können:

```
This script is invoked by /usr/lpp/X11/bin/xinit
.
.
.
#*****
Start the X clients. Change the following lines to *
whatever command(s) you desire! *
The default clients are an analog clock (xclock), an lft *
terminal emulator (aixterm), and the Motif Window Manager *
(mwm). *
#*****
exec mwm
```

*Datei .Xdefaults:*

Wenn Sie mit einer AIXwindows-Schnittstelle arbeiten, können Sie diese Schnittstelle mit der Datei `.Xdefaults` anpassen. In AIXwindows können Sie Vorgaben für visuelle Merkmale, wie z. B. Farbeinstellungen und Schriftarten, festlegen.

Viele Aspekte einer auf dem Betriebssystem Windows basierenden Anwendung in Bezug auf die Darstellung und Funktionsweise werden von einer Gruppe von Variablen gesteuert, die als *Ressourcen* bezeichnet werden. Das Erscheinungsbild und die Funktionsweise einer Ressource werden durch den ihr zugeordneten Wert bestimmt. Es gibt verschiedene Arten von Werten für Ressourcen. Den Ressourcen, die z. B. die Farbeinstellungen steuern, können vordefinierte Werte, wie *DarkSlateBlue* oder *Black* zugeordnet werden. Den Ressourcen, die Abmessungen angeben, werden numerische Werte zugeordnet. Manche Ressourcen verwenden die booleschen Werte *true* und *false*.

Falls Sie keine Datei `.Xdefaults` in Ihrem Ausgangsverzeichnis haben, können Sie diese mit jedem Texteditor erstellen. Sobald diese Datei in Ihrem Ausgangsverzeichnis angelegt ist, können Sie in dieser Datei Ihre Ressourcenwerte festlegen. Im Verzeichnis `/usr/lpp/X11/defaults` befindet sich eine Beispielstandarddatei mit dem Namen `Xdefaults.tpl`.

Das folgende Beispiel enthält einen Auszug aus einer typischen Datei `.Xdefaults`:

```
*AutoRaise: on
*DeIconifyWarp: on
*warp: on
*TitleFont: andysans12
*scrollBar: true
*font: Rom10.500
Mwm*menu*foreground: black
Mwm*menu*background: CornflowerBlue
Mwm*menu*RootMenu*foreground: black
Mwm*menu*RootMenu*background: CornflowerBlue
Mwm*icon*foreground: grey25
Mwm*icon*background: LightGray
Mwm*foreground: black
Mwm*background: LightSkyBlue
Mwm*bottomShadowColor: Blue1
Mwm*topShadowColor: CornflowerBlue
Mwm*activeForeground: white
Mwm*activeBackground: Blue1
Mwm*activeBottomShadowColor: black
Mwm*activeTopShadowColor: LightSkyBlue
Mwm*border: black
Mwm*highlight: white
```

```

aixterm.foreground: green
aixterm.background: black
aixterm.fullcursor: true
aixterm.ScrollKey: on
aixterm.autoRaise: true
aixterm.autoRaiseDelay: 2
aixterm.boldFont: Rom10.500
aixterm.geometry: 80x25
aixterm.iconFont: Rom8.500
aixterm.iconStartup: false
aixterm.jumpScroll: true
aixterm.reverseWrap: true
aixterm.saveLines: 500
aixterm.scrollInput: true
aixterm.scrollKey: false
aixterm.title: AIX

```

*Datei .mwmrc:*

Die meisten Funktionen können mit den Ressourcen in der Datei `.Xdefaults` angepasst werden. Tastenbelegungen, Maustastenbelegungen und Menüdefinitionen für den Fenstermanager werden jedoch in der Ergänzungsdatei `.mwmrc` definiert, die von Ressourcen in der Datei `.Xdefaults` verwendet wird.

Ist die Datei `.mwmrc` nicht im Ausgangsverzeichnis vorhanden, kann sie wie folgt kopiert werden:

```
cp /usr/lib/X11/system.mwmrc .mwmrc
```

Da die Datei `.mwmrc` die globale Wirkung der Datei `system.mwmrc` außer Kraft setzt, entstehen keine Konflikte mit den Angaben anderer Benutzer.

Das folgende Beispiel enthält einen Auszug aus einer typischen Datei `system.mwmrc`:

```

DEFAULT mwm RESOURCE DESCRIPTION FILE (system.mwmrc)
#
menu pane descriptions
#
Root Menu Description
Menu RootMenu
{ "Root Menu" f.title
 no-label f.separator
 "New Window" f.exec "aixterm &"
 "Shuffle Up" f.circle_up
 "Shuffle Down" f.circle_down
 "Refresh" f.refresh
 no-label f.separator
 "Restart" f.restart
 "Quit" f.quit_mwm
}
Default Window Menu Description

Menu DefaultWindowMenu MwmWindowMenu
{ "Restore" _R Alt<Key>F5 f.normalize
 "Move" _M Alt<Key>F7 f.move
 "Size" _S Alt<Key>F8 f.resize
 "Minimize" _n Alt<Key>F9 f.minimize
 "Maximize" _x Alt<Key>F10 f.maximize
 "Lower" _L Alt<Key>F3 f.lower
 no-label f.separator
 "Close" _C Alt<Key>F4 f.kill
}
no acclerator window menu
Menu NoAccWindowMenu
{
 "Restore" _R f.normalize
 "Move" _M f.move

```

```

"Size" _S f.resize
"Minimize" _n f.minimize
"Maximize" _x f.maximize
"Lower" _L f.lower
no-label _ f.separator
"Close" _C f.kill
}

Keys DefaultKeyBindings
{
 Shift<Key>Escape icon|window f.post_wmenu
 Meta<Key>space icon|window f.post_wmenu
 Meta<Key>Tab root|icon|window f.next_key
 Meta Shift<Key>Tab root|icon|window f.prev_key
 Meta<Key>Escape root|icon|window f.next_key
 Meta Shift<Key>Escape root|icon|window f.prev_key
 Meta Ctrl Shift<Key>exclam root|icon|window f.set_behavior
}

#
button binding descriptions
#

Buttons DefaultButtonBindings
{
 <Btn1Down> frame|icon f.raise
 <Btn3Down> frame|icon f.post_wmenu
 <Btn1Down> root f.menu RootMenu
 <Btn3Down> root f.menu RootMenu
 Meta<Btn1Down> icon|window f.lower
 Meta<Btn2Down> window|icon f.resize
 Meta<Btn3Down> window f.move
}

Buttons PointerButtonBindings
{
 <Btn1Down> frame|icon f.raise
 <Btn2Down> frame|icon f.post_wmenu
 <Btn3Down> frame|icon f.lower
 <Btn1Down> root f.menu RootMenu
 Meta<Btn2Down> window|icon f.resize
 Meta<Btn3Down> window|icon f.move
}

#
END OF mwm RESOURCE DESCRIPTION FILE
#

```

### Shellvariablen exportieren (Shellbefehl export):

Bei einer *lokalen* Shellvariablen handelt es sich um eine Variable, die ausschließlich der Shell bekannt ist, mit der sie erstellt wurde. Beim Öffnen einer neuen Shell werden die Variablen der alten Shell nicht für die neue Shell übernommen. Wenn die neuen Shells, die Sie öffnen, die Variablen einer alten Shell verwenden sollten, können Sie die Variablen exportieren, um sie *verfügbar* zu machen.

Mit dem Befehl **export** können Sie lokale Variablen verfügbar machen. Um die lokalen Shellvariablen automatisch verfügbar zu machen, müssen sie in die Datei `.profile` exportiert werden.

**Anmerkung:** Variablen können nur in untergeordnete Shells, aber nicht in übergeordnete Shells exportiert werden.

Beispiele:

- Geben Sie Folgendes ein, wenn Sie aus der lokalen Shellvariablen `PATH` eine globale Variable machen möchten:  
`export PATH`
- Geben Sie Folgendes ein, um alle exportierten Variablen aufzulisten:

export

Das System zeigt Informationen ähnlich den folgenden an:

```
DISPLAY=unix:0
EDITOR=vi
ENV=$HOME/.env
HISTFILE=/u/denise/.history
HISTSIZE=500
HOME=/u/denise
LANG=En_US
LOGNAME=denise
MAIL=/usr/mail/denise
MAILCHECK=0
MAILMSG=**YOU HAVE NEW MAIL.
USE THE mail COMMAND TO SEE YOUR MAILPATH=/usr/mail/denise?denise has mail !!!
MAILRECORD=/u/denise/.Outmail
PATH=/usr/ucb:/usr/lpp/X11/bin:/bin:/usr/bin:/etc:/u/denise:/u/denise/bin:/u/bin1
PWD=/u/denise
SHELL=/bin/ksh
```

### Standardschriftart ändern (Befehl **chfont**):

Mit dem Befehl **chfont** oder dem Befehl **smit** können Sie die beim Systemstart verwendete Schriftgröße ändern. Eine *Schriftartpalette* ist eine Datei, die vom System für die Definition und Erkennung der verfügbaren Schriftarten verwendet wird.

**Anmerkung:** Sie müssen zum Ausführen des Befehls **chfont** Rootberechtigung besitzen.

#### Befehl **chfont**

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **chfont**:

- Geben Sie Folgendes ein, um die aktive Schriftart durch die fünfte Schriftart in der Schriftartpalette zu ersetzen:

```
chfont -a5
```

- Geben Sie Folgendes ein, um die Schriftart in Roman, kursiv und fett mit gleicher Größe zu ändern:

```
chfont -n /usr/lpp/fonts/It114.snf /usr/lpp/fonts/Bld14.snf /usr/lpp/fonts/Rom14.snf
```

Die vollständige Syntax des Befehls **chfont** ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

#### Befehl **smit**

Der Befehl **chfont** kann auch mit **smit** ausgeführt werden.

Geben Sie Folgendes ein, um die aktive Schriftart auszuwählen:

```
smit chfont
```

Geben Sie Folgendes ein, um die Schriftartpalette auszuwählen:

```
smit chfontpl
```

### Steuertasten ändern (Befehl **stty**):

Verwenden Sie zum Ändern der Tasten, die das Terminal als Steuertasten verwendet, den Befehl **stty**.

Die Änderungen an den Steuertasten bleiben bis zum Abmelden vom System aktiv. Um die Änderungen als permanente Einstellungen zu übernehmen, müssen sie in der Datei `.profile` angegeben werden.

Beispiele:

- Geben Sie Folgendes ein, um die Tastenkombination Strg-Z als Unterbrechungstaste zu definieren:

```
stty intr ^Z
```

Zwischen `intr` und `^Z` muss unbedingt ein Leerzeichen eingefügt werden.

- Geben Sie Folgendes ein, um alle Steuertasten auf die Standardwerte zurückzusetzen:

```
stty sane
```

- Geben Sie Folgendes ein, um die aktuellen Einstellungen anzuzeigen:

```
stty -a
```

Die vollständige Syntax des Befehls **stty** ist in der Veröffentlichung *Commands Reference, Volume 5* beschrieben.

### Eingabeaufforderung ändern:

Sie können die Eingabeaufforderung ändern.

Die Shell verwendet die folgenden Variablen für die Eingabeaufforderung:

Eintrag	Beschreibung
<code>PS1</code>	Normale Eingabeaufforderung.
<code>PS2</code>	Eingabeaufforderung, die verwendet wird, wenn von der Shell weitere Eingaben erwartet werden.
<code>PS3</code>	Eingabeaufforderung in Sitzungen mit Rootberechtigung.

Sie können die Zeichen für Eingabeaufforderungen ändern, indem Sie den Wert der zugehörigen Shellvariablen ändern. Die Änderungen an den Eingabeaufforderungen bleiben bis zum Abmelden von der Sitzung aktiv. Um die Änderungen als permanente Einstellung zu übernehmen, müssen Sie sie in der Datei `.env` angeben.

Beispiele:

- Geben Sie Folgendes ein, um den aktuellen Wert der Variablen `PS1` anzuzeigen:

```
echo "prompt is $PS1"
```

Das System zeigt Informationen ähnlich den folgenden an:

```
prompt is $
```

- Geben Sie Folgendes ein, um die Eingabeaufforderung in `Ready>` zu ändern:

```
PS1="Ready> "
```

- Geben Sie Folgendes ein, um die Eingabeaufforderung zur Fortsetzung der Eingabe in `Weiter->` zu ändern:

```
PS2="Weiter->"
```

- Geben Sie Folgendes ein, um die Rooteingabeaufforderung in `Root->` zu ändern:

```
PS3="Root-> "
```

## Referenz für BSD-Systeme

Dieser Anhang ist für Systemadministratoren bestimmt, die mit den UNIX-Betriebssystemen BSD 4.3 oder System V vertraut sind. In diesen Informationen werden die Unterschiede und Ähnlichkeiten zwischen diesen Systemen und AIX erläutert.

Lesen Sie die folgenden Abschnitte.

### BSD-Konzepte

Bevor Sie beginnen, mit Berkeley Software Distribution (BSD) zu arbeiten, müssen Sie sich mit den Unterschieden zwischen BSD und AIX vertraut machen.

## Einführung in AIX für BSD-Systemmanager:

Die folgenden Tipps sind für BSD-Systemmanager (Berkeley Software Distribution) als Einführung in die Verwaltung von AIX bestimmt.

- Melden Sie sich zunächst als Root an der grafisch orientierten Konsole an.
- Verwenden Sie für die Systemverwaltung die Systemkonsole so lange, bis Sie mit dem System vertraut sind. Es ist einfacher für Sie, mit der Systemkonsole als mit einem fernen Terminal zu arbeiten. Sobald Sie mit dem System vertraut sind, können Sie über Remote-Zugriff von einem xterm oder einem ASCII-Terminal aus arbeiten.
- Nutzen Sie für die Systemverwaltungstasks die zahlreichen Funktionen von AIX, z. B.:
  - System Management Interface Tool (SMIT). SMIT stellt eine Schnittstelle zwischen Systemmanagern und den Konfigurations- und Verwaltungsbefehlen dar. SMIT kann Systemmanager bei den meisten Systemverwaltungstasks unterstützen.
  - Object Data Manager (ODM). ODM stellt Routinen bereit, die auf Objekte in den ODM-Datenbanken zugreifen. Die ODM-Datenbanken enthalten Einheitenkonfigurationsdaten.
  - System Resource Controller (SRC). SRC unterstützt den Zugriff auf und die Steuerung von Dämonprozessen und anderen Systemressourcen in einer einzigen Schnittstelle.

### Zugehörige Konzepte:

„Eine große Anzahl von Einheiten konfigurieren“ auf Seite 411

Zu Einheiten gehören Hardwarekomponenten wie Drucker, Laufwerke, Adapter, Busse und Platteneinschübe sowie Pseudoeinheiten wie die Fehlergerätedatei und die Nullgerätedatei. Einheitentreiber sind im Verzeichnis `/usr/lib/drivers` gespeichert.

„System Resource Controller“ auf Seite 192

Der System Resource Controller (SRC) stellt eine Reihe von Befehlen und Subroutinen zur Verfügung, die dem Systemmanager und dem Programmierer die Erstellung und Steuerung von Subsystemen erleichtern.

### Hauptunterschiede zwischen BSD 4.3 und AIX:

Im Folgenden finden Sie eine Übersicht über die Hauptunterschiede zwischen den Systemen AIX und BSD 4.3.

In AIX werden die Netzdämonprozesse über die Datei `/etc/rc.tcpip` und nicht über die Datei `/etc/rc.local` gestartet. Das Shell-Script `/etc/rc.tcpip` wird über die Datei `/etc/inittab` und nicht über die Datei `/etc/rc` aufgerufen.

Wenn System Resource Controller (SRC) aktiv ist, werden die TCP/IP-Dämonprozesse unter SRC-Steuerung ausgeführt. Wenn die TCP/IP-Dämonprozesse nicht unter SRC-Steuerung ausgeführt werden sollen, stellen Sie das System mit dem Direktaufruf **smit setbootup\_option** auf die **rc**-Konfiguration im BSD-Stil um.

Die folgenden, in BSD 4.3 verfügbaren Netzverwaltungsfunktionen werden von AIX unterstützt:

- SYSLOG-Protokolleinrichtungen auf Kernel-Level
- Zugriffsberechtigungen für UNIX-Domänen-Sockets

### Speicherung von Konfigurationsdaten

In BSD 4.3 werden die Konfigurationsdaten gewöhnlich in ASCII-Dateien gespeichert. Zugehörige Einzelinformationen werden in derselben Zeile gespeichert. Operationen zur Bearbeitung von Datensätzen (z. B. Sortieren und Suchen) können für die ASCII-Datei selbst ausgeführt werden. Die Datensätze können in der Länge variieren und werden durch einen Zeilenvorschub beendet. BSD 4.3 stellt Tools bereit, mit denen einige potenziell große ASCII-Dateien in ein Datenbankformat

(dbm) konvertiert werden können. Die entsprechenden Bibliotheksfunktionen verwenden das dbm-Dateipaar, sofern dieses vorhanden ist. Sind die dbm-Dateien nicht vorhanden, wird die ursprüngliche ASCII-Datei verwendet.

Einige Konfigurationsdaten für AIX werden in ASCII-Dateien, aber häufig in einem *Zeilengruppenformat* gespeichert. Eine Zeilengruppe ist eine Gruppe zusammengehöriger Einzelinformationen, die in einer Gruppe von mehreren Zeilen gespeichert werden. Jede Einzelinformation hat einen Kennsatz, um den Inhalt der Datei verständlicher zu machen.

AIX unterstützt auch dbm-Versionen von Kennwort- und Benutzerdaten. Die Dateien `/etc/passwd`, `/etc/group` und `/etc/inittab` sind Beispiele für Dateien in AIX, in denen die Daten in traditionellem und nicht im Zeilengruppenformat gespeichert werden.

Weitere Konfigurationsdaten für AIX werden in Dateien gespeichert, die vom Object Data Manager (ODM) verwaltet werden. Mit System Management Interface Tool (SMIT) können Daten in ODM-Dateien bearbeitet und angezeigt werden. Alternativ können Sie zum Anzeigen dieser Dateien die ODM-Befehle direkt verwenden. Verwenden Sie zum Abfragen der ODM-Dateien die folgenden Befehle:

- **odmget**
- **odmshow.**

Mit den folgenden ODM-Befehlen können ODM-Dateien geändert werden:

- **odmadd**
- **odmcreate**
- **odmdrop**
- **odmchange**
- **odmdelete.**

**Achtung:** Wenn beim Ändern von ODM-Dateien Fehler gemacht werden, können diese zu einem Systemfehler führen und möglicherweise einen erfolgreichen Neustart des Systems verhindern. Verwenden Sie die ODM-Befehle für ODM-Dateien nur direkt, wenn taskspezifische Befehle, wie z. B. die von SMIT generierten, nicht erfolgreich sind.

## Konfigurationsverwaltung

Beim Starten eines Systems, auf dem AIX ausgeführt wird, werden eine Reihe von konfigurationsspezifischen Befehlen vom Konfigurationsmanager aufgerufen. Diese konfigurationsspezifischen Befehle werden *Methoden* genannt. Methoden identifizieren die Einheiten im System und aktualisieren die entsprechenden ODM-Dateien im Verzeichnis `/etc/objrepos`.

Die Einheitengerätedateien in `/dev` sind nicht vorinstalliert. Einige Gerätedateien, wie z. B. die für Festplatten, werden automatisch während des Konfigurationsprozesses beim Systemstart erstellt. Andere Gerätedateien wie die für ASCII-Terminals müssen vom Systemadministrator mit dem SMIT-Menü **Einheiten** erstellt werden. Diese Informationen werden für spätere Verwendung durch das System im ODM aufbewahrt.

## Plattenverwaltung

In AIX werden Plattenlaufwerke als *physische Datenträger* bezeichnet und Partitionen als *logische Datenträger*. Wie in BSD 4.3 kann ein einzelner physischer Datenträger mehrere logische Datenträger haben. Anders als in BSD 4.3 kann sich jedoch in AIX ein einzelner logischer Datenträger über mehrere physische Datenträger erstrecken. Hierfür müssen Sie mehrere physische Datenträger zu einer *Datenträgergruppe* zusammenfassen und in der Datenträgergruppe logische Datenträger erstellen.

Für die Verwaltung von Dateisystemen und Datenträgern werden in AIX unter anderem die folgenden Befehle verwendet:

- **crfs**
- **varyonvg**

- **varyoffvg**
- **lsvg**
- **importvg**
- **exportvg**.

Die folgenden BSD-4.3-Befehle sind ebenfalls verfügbar:

- **mkfs**
- **fsck**
- **fsdb**
- **mount**
- **umount**.

Die Unterschiede zwischen diesen Befehlen in BSD 4.3 und AIX sind im Abschnitt „Dateisysteme für BSD-4.3-Systemmanager“ auf Seite 361 beschrieben.

BSD 4.3 verwaltet eine Liste mit Dateisystemen in der Datei `/etc/fstab`. AIX verwaltet für jedes Dateisystem eine Zeilengruppe in der Datei `/etc/filesystems`.

### **Befehl tn3270**

Der Befehl **tn3270** ist eine Verbindung zum Befehl **telnet**, verwendet aber die Datei `/etc/map3270` und die aktuelle Umgebungsvariable `TERM` für die Bereitstellung der 3270-Tastaturbelegungen. Deshalb arbeitet der Befehl **tn3270** genauso wie die BSD-Version.

Wenn Sie die von den Befehlen **tn3270**, **telnet** und **tn** verwendeten Standardwerte für die Escape-Zeichenfolgen ändern möchten, setzen Sie die Umgebungsvariable `TNESC`, bevor Sie diese Befehle ausführen.

### **Neue Befehle**

Für die Unterstützung neuer Konfigurations- und Plattenverwaltungssysteme stellt AIX ungefähr 150 Befehle zur Verfügung, die für BSD-4.3-Administratoren neu sind.

### **Systemstart**

AIX unterstützt die automatische Identifizierung und Konfiguration von Einheiten. Deshalb weicht der Startprozess erheblich von dem auf BSD-4.3-Systemen ab. Zusätzlich zum Kernel werden ein Image eines Bootdateisystems und die vorherigen Basiseinheitenkonfigurationsdaten auf eine RAM-Platte geladen. In der ersten Phase des Systemstarts werden ausreichend Konfigurationsdaten geladen und geprüft, um den Zugriff auf logische Datenträger zu ermöglichen. Die Einheit für den Paging-Bereich wird vom Kernel identifiziert, und das Stammdateisystem der Festplatte wird geprüft. Zu diesem Zeitpunkt wechselt das Betriebssystem vom RAM-Stammdateisystem auf die Festplatte und schließt die Startprozedur, einschließlich der Konfiguration weiterer Einheiten ab.

### **Benutzerberechtigung**

In BSD 4.3 und den Versionen der AT&T-UNIX-Betriebssysteme vor SVR4 werden alle Benutzerauthentifizierungsdaten, einschließlich verschlüsselter Kennwörter, in der Datei `/etc/passwd` gespeichert. Traditionell kann die Datei `/etc/passwd` von allen gelesen werden.

Auf SVR4-Systemen werden verschlüsselte Kennwörter aus der Datei `/etc/passwd` entfernt und in der Datei `/etc/shadow` gespeichert. Nur Benutzer mit Rootberechtigung und anerkannte Programme (z. B. `/bin/login`) können die Datei `/etc/shadow` lesen.

AIX speichert verschlüsselte Kennwörter in der Datei `/etc/security/passwd`. Weitere Dateien im Verzeichnis `/etc/security` sind `user` und `limits`. Diese drei Dateien definieren die Methoden, mit denen ein Benutzer auf das System zugreifen kann (z. B. mit den Befehlen **rlogin** und **telnet**), und die Ressourcengrenzen des Benutzers (z. B. Dateigröße und Adressraum).

### **Drucken**



Die meisten BSD-4.3-Druckbefehle werden mit geringfügigen Abweichungen unterstützt. Ein Unterschied besteht darin, dass in AIX die Datei /etc/qconfig die Konfigurationsdatei ist.

Das Zeilendrucksystem von AIX kann mit dem Zeilendrucksystem von BSD 4.3 interagieren, so dass Druckjobs an und von BSD-4.3-Systemen übergeben werden können.

## Shells

AIX unterstützt die Bourne-Shell, die C-Shell und die Korn-Shell. Der vollständige Pfadname für das Bourne-Shellprogramm ist /bin/bsh. Die Datei /bin/sh ist eine feste Verbindung zur Datei /bin/ksh. Diese Datei kann vom Administrator geändert werden.

**setuid** und **setgid** für Shell-Scripts werden von AIX in keiner Shell unterstützt.

### Anmerkung:

1. AIX besitzt keine Shell-Scripts, die sich auf /bin/sh stützen. Viele Shell-Scripts von anderen Systemen stützen sich jedoch auf /bin/sh (die Bourne-Shell).
2. Obwohl sich die Bourne-Shell und die Korn-Shell sehr ähnlich sind, ist die Korn-Shell kein vollkommenes Superset der Bourne-Shell.

### Zugehörige Verweise:

„Systemverwaltungsbefehle für die Systemmanager von BSD 4.3“ auf Seite 357

Diese Liste enthält Befehle, die speziell für die Verwaltung der Umgebung für AIX bestimmt sind.

### Dateivergleichstabelle für BSD 4.3, SVR4 und AIX:

In der folgenden Tabelle werden Dateinamen und Funktionen in BSD 4.3, SVR4 und AIX verglichen.

Tabelle 61. Dateivergleichstabelle

Datei in BSD 4.3	Datei in SVR4	Datei in AIX	Datenbank	Typ (odm/dbm)
L-Devices	Devices	Devices	Nein	
L-dialcodes	Dialcodes	Dialcodes	Nein	
L.cmds	Permissions	Permissions	Nein	
L.sys	Systems	Systems	Nein	
USERFILE	Permissions	Permissions	Nein	
aliases	mail/namefiles	aliases	aliasesDB/DB	dbm
fstab	vfstab	filesystems	Nein	
ftpusers	ftpusers	ftpusers	Nein	
gettytab		Nicht zutreffend		
group	group	group	Nein	
hosts	hosts	hosts	Nein	
hosts.equiv	hosts.equiv	hosts.equiv	Nein	
inetd.conf	inetd.conf	inetd.conf	Nein	
map3270	Nicht zutreffend	map3270	Nein	
motd	motd	motd	Nein	
mtab	mnttab	Nicht zutreffend	Nein	
named.boot	named.boot	named.boot	Nein	
named.ca		named.ca	Nein	
named.hosts		named.data (siehe Anmerkung)	Nein	
named.local		named.local	Nein	
named.pid	named.pid	named.pid	Nein	
named.rev		named.rev	Nein	
networks	networks	networks	Nein	

Tabella 61. Dateivergleichstabelle (Forts.)

Datei in BSD 4.3	Datei in SVR4	Datei in AIX	Datenbank	Typ (odm/dbm)
passwd	passwd	passwd	Nein	
printcap	qconfig	qconfig		
protocols		protocols	Nein	
remote	remote	remote	Nein	
resolv.conf	resolv.conf	resolv.conf	Nein	
sendmail.cf	sendmail.cf	sendmail.cf	sendmail.cfDB	Nicht zutreffend
services		services	Nein	
shells	shells	Nicht zutreffend		
stab		Nicht zutreffend		
syslog.conf		syslog.conf	Nein	
syslog.pid		syslog.pid	Nein	
termcap	terminfo	terminfo		
ttys	ttys	Nicht zutreffend	Ja	odm
types		Nicht zutreffend		
utmp	utmp	utmp		
vfont		Nicht zutreffend		
vgrindefs		vgrindefs		
wtmp	wtmp	wtmp		

**Anmerkung:** Die Dateinamen `named.ca`, `named.hosts`, `named.local` und `named.rev` können in der Datei `named.boot` vom Benutzer definiert werden. Hier werden jedoch die Namen, die für diese Dateien in der Dokumentation zu AIX angegeben sind, verwendet.

### Namens- und Adressauflösung:

Die Subroutinen **gethostbyname** und **gethostbyaddr** in der Bibliothek `libc` bieten Unterstützung für DNS (Domain Name Service), Network Information Services (NIS, früher Yellow Pages) und die Datenbank `/etc/hosts`.

Wenn die Datei `/etc/resolv.conf` vorhanden ist, wird zuerst der Namensserver geprüft. Wenn der Name nicht aufgelöst wird und NIS aktiv ist, wird NIS geprüft. Wenn NIS nicht aktiv ist, wird die Datei `/etc/hosts` geprüft.

### Onlinedokumentation und Befehl `man` für BSD-4.3-Systemmanager:

AIX unterstützt die Befehle **man -k**, **apropos** und **whatis**, aber die von diesen Befehlen verwendete Datenbank muss zuerst mit dem Befehl **catman -w** erstellt werden.

Der Befehl **man** sucht zuerst nach unstrukturierten Textseiten in den Dateien `/usr/man/cat?`. Anschließend sucht er nach Seiten im **nroff**-Format in den Dateien `/usr/man/man?`. Neue Man-Pages können in unstrukturiertem Textformat oder im **nroff**-Format hinzugefügt werden.

### Anmerkung:

- Die Textseiten des Befehls **man** werden nicht mit dem System bereitgestellt. Der Befehl **catman** erstellt aus diesen Textseiten die Datenbank. Bei diesen Seiten kann es sich entweder um unstrukturierte Textseiten, die in den Dateien `/usr/man/cat?` gespeichert sind, oder um Seiten im **nroff**-Format, die in den Dateien `/usr/man/man?` gespeichert sind, handeln.
- Das Lizenzprogramm für Textformatierung muss installiert werden, damit der Befehl **nroff** für den Befehl **man** verfügbar ist, wenn dieser Man-Pages im **nroff**-Format lesen muss.

Weitere Informationen finden Sie in den Beschreibungen der Befehle **man**, **apropos**, **whatis** und **catman** in der Veröffentlichung "Commands Reference".

### **NFS und NIS (früher Yellow Pages) für BSD-4.3-Systemmanager:**

Im Folgenden werden NFS und NIS für BSD-4.3-Systemmanager beschrieben.

NFS- (Network File System) und NIS-Dämonprozesse (Network Information Services) werden über die Datei `/etc/rc.nfs` gestartet. Damit die NFS- und NIS-Dämonprozesse jedoch gestartet werden können, muss der Dämon **portmap** in der Datei `/etc/rc.tcpip` gestartet werden. Standardmäßig wird die Datei `/etc/rc.nfs` nicht von der Datei `/etc/inittab` aufgerufen. Wenn Sie der Datei `/etc/inittab` eine Zeile für den Aufruf des Scripts `/etc/rc.nfs` hinzufügen, muss dieses Script nach dem Script `/etc/rc.tcpip` aufgerufen werden.

Wenn NIS aktiv ist, fügen Sie einen Stammeintrag vor dem Eintrag `+::` (Pluszeichen, Doppelpunkt, Doppelpunkt) in der Datei `/etc/passwd` und einen Systemeintrag vor dem Eintrag `+::` in der Datei `/etc/group` ein. Dies ermöglicht einem Systemadministrator, sich als Root anzumelden und Änderungen vorzunehmen, falls das System nicht mit dem NIS-Server kommunizieren kann.

NFS kann mit dem SMIT-Direktaufruf **smit nfs** konfiguriert werden. In den Menüs von SMIT bezieht sich NIS auf Network Information Services (früher Yellow Pages). Viele NFS- und NIS-Befehle befinden sich in den Verzeichnissen `/etc` und `/usr/etc`.

Einige NFS-Umgebungen verwenden den Befehl **arch**, um Maschinenfamilien und Maschinentypen zu identifizieren. Wenn Sie beispielsweise eine Maschine des Typs IBM<sup>®</sup> RS/6000 verwenden, geben Sie die Kennung **power** für die Familie (CPU) und die Kennung **ibm6000** für den Typ (Maschine) an.

### **Benutzerkennwörter für BSD-4.3-Systemmanager:**

Wenn Sie den Befehl `/bin/passwd` für AIX als Root ausführen, werden Sie aufgefordert, das aktuelle Kennwort von Root einzugeben.

Im Folgenden sehen Sie ein Beispiel für die Verwendung des Befehls `/bin/passwd`:

```
passwd cslater
Changing password for "cslater"
Enter root's Password or
cslater's Old password:
cslater's New password:
Re-enter cslater's
new password:
#
```

Die BSD-4.3-Version fordert Sie nicht zur Eingabe des aktuellen Kennworts von Root auf. Im Folgenden sehen Sie ein Beispiel für die Verwendung der BSD-4.3-Version:

```
passwd cslater
New password:
Retype new password:
#
```

### **BSD verwalten**

Es sind mehrere Befehle für BSD verfügbar, die Sie für die Ermittlung der Systemleitung, die Ausführung von Druckoperationen und die Verwaltung des Systems verwenden können.

### **Abrechnung für BSD-4.3-Systemmanager:**

Die Abrechnungsdateien im Verzeichnis `/usr/lib/acct` und die Berichtstools für Systemaktivitäten im Verzeichnis `/usr/lib/sa` für AIX sind identisch mit denen, die in AT&T System V Release 4 (SVR4) enthalten sind. Zusätzlich sind die Abrechnungsdienstprogramme für BSD 4.3 verfügbar.

Viele Abrechnungsbefehle befinden sich im Verzeichnis `/usr/lib/acct`. Verwenden Sie zum Starten der Systemabrechnung den Befehl `/usr/lib/acct/startup`. Wenn die Abrechnung nicht gestartet ist, können Befehle wie **lastcomm(1)** keine Informationen zurückgeben.

AIX stellt die folgenden Abrechnungsfunktionen aus BSD 4.3 bereit:

Eintrag	Beschreibung
<b>last(1)</b>	Gibt die letzten Anmeldungen für Benutzer und Terminals an.
<b>lastcomm(1)</b>	Zeigt die zuletzt ausgeführten Befehle in umgekehrter Reihenfolge an.
<b>acct(3)</b>	Aktiviert und inaktiviert die Prozessabrechnung.
<b>ac(8)</b>	Anmeldungsabrechnung.
<b>accton(8)</b>	Aktiviert und inaktiviert die Systemabrechnung.
<b>sa(8)</b>	Ist für die generelle Verwaltung der Systemabrechnungsdateien bestimmt.

AIX unterstützt außerdem die folgenden Abrechnungsbefehle und Bibliotheksfunktionen von System V Interface Definition (SVID) Issue II:

Eintrag	Beschreibung
<b>acctcms(1)</b>	Erzeugt aus Abrechnungsdatensätzen Befehlsbenutzungsübersichten.
<b>acctcom(1)</b>	Zeigt ausgewählte Übersichten über Prozessabrechnungssätze an.
<b>acctcon1(1)</b>	Konvertiert Anmelde-/Abmeldesätze in Sitzungssätze.
<b>acctcon2(1)</b>	Konvertiert Anmelde-/Abmeldesätze in Gesamtabrechnungssätze.
<b>acctdisk(1)</b>	Generiert aus der Ausgabe des Befehls <b>diskusg(1)</b> Gesamtabrechnungssätze.
<b>acctmerg(1)</b>	Fügt Gesamtabrechnungsdateien in einer temporären Datei zusammen.
<b>accton(1)</b>	Aktiviert die Abrechnung.
<b>acctprc1(1)</b>	Verarbeitet Abrechnungsdaten des Befehls <b>acct(3)</b> .
<b>acctprc2(1)</b>	Verarbeitet die Ausgabe des Befehls <b>acctprc1(1)</b> zu Gesamtabrechnungssätzen.
<b>acctwtmp(1)</b>	Bearbeitet Abrechnungssätze zur Verbindungsdauer.
<b>chargefee(1)</b>	Berechnet dem Anmeldenamen Gebühren.
<b>ckpacct(1)</b>	Überprüft die Größe der Datei <code>/usr/adm/pacct</code> .
<b>diskusg(1)</b>	Generiert Abrechnungsdaten zur Plattenbelegung.
<b>dodisk(1)</b>	Führt die Plattenabrechnung durch.
<b>fwtmp(1)</b>	Konvertiert binäre Datensätze (Datei <code>wtmp</code> ) in formatierte ASCII-Datensätze.
	<b>Anmerkung:</b> Die Datei <code>wtmp</code> befindet sich im Verzeichnis <code>/var/adm</code> .
<b>lastlogin(1)</b>	Aktualisiert das letzte Datum, an dem sich die einzelnen Personen jeweils angemeldet haben.
<b>monacct(1)</b>	Erstellt Monatsübersichtsdateien.
<b>prctmp(1)</b>	Gibt die vom Befehl <b>acctcon1(1)</b> erzeugte Sitzungssatzdatei aus.
<b>prdaily(1)</b>	Formatiert einen Bericht über die Abrechnungsdaten des Vortags.
<b>pracct(1)</b>	Formatiert und gibt eine Gesamtabrechnungsdatei aus.
<b>runacct(1)</b>	Führt die tägliche Abrechnung durch.
<b>shutacct(1)</b>	Wird von der Systemabschlussprozedur aufgerufen, um die Abrechnung zu stoppen und die Ursache zu protokollieren.
<b>startup(1)</b>	Wird von der Systeminitialisierungsprozedur aufgerufen, um die Abrechnung zu starten.
<b>turnacct(1)</b>	Aktiviert und inaktiviert die Prozessabrechnung.
<b>wtmpfix(1)</b>	Korrigiert Zeit- und Datumsmarken in einer Datei mit dem Format <code>wtmp</code> .

### Sicherung für BSD-4.3-Systemmanager:

BSD-4.3-Systemmanager können Daten sichern.

Mit den Befehlen **tar** und **cpio** können Sie Daten zwischen Systemen verschieben. Der Befehl **tar** für AIX ist nicht vollständig kompatibel mit dem BSD-4.3-Befehl **tar**. Der Befehl **tar** für AIX erfordert das Flag **-B** (geblockte Eingabe), wenn er aus einer Befehlskette liest. Der AT&T-Befehl **cpio** ist mit dieser Version kompatibel.

AIX kann die Formate der Befehle **dump** und **restore** lesen und schreiben. Beispielsweise ist der Befehl **backup** für AIX mit der Syntax

`backup -0uf Einheit Name_des_Dateisystems`

identisch mit dem BSD-4.3-Befehl **dump** mit der folgenden Syntax:

`dump 0uf Einheit Name_des_Dateisystems`

Gleichermaßen ist der Befehl **restore** für AIX mit der Syntax

`restore -mivf Einheit`

identisch mit dem BSD-4.3-Befehl **restore** mit der folgenden Syntax:

`restore ivf Einheit`

AIX unterstützt auch die BSD-4.3-Befehle **rdump** und **rrestore**. Der einzige Unterschied zwischen den beiden Versionen besteht darin, dass für AIX jedem Argument ein Minuszeichen (-) vorangestellt werden muss. Der Befehl

`rdump -0 -f orca:/dev/rmt0 /dev/hd2`

entspricht dem folgenden BSD-4.3-Befehl:

`rdump 0f orca:/dev/rmt0 /dev/hd2`

Der Befehl **backup** für AIX mit der Syntax

`backup -0f /dev/rmt0 /dev/hd2`

entspricht dem BSD-4.3-Befehl **dump** mit der folgenden Syntax:

`dump 0f /dev/rmt0 /dev/hd2`

### **Unterstützung für Bandlaufwerke, die keine IBM SCSI-Bandlaufwerke sind**

AIX bietet keine direkte Unterstützung für Bandlaufwerke, die keine IBM SCSI-Bandlaufwerke sind. Sie können jedoch einen eigenen Header und eine eigene Schnittstelle hinzufügen, die den IBM SCSI-Treiber verwendet.

#### **Zugehörige Konzepte:**

„Systemsicherung“ auf Seite 22

Die nächste Überlegung nach der Inbetriebnahme des Systems muss der Sicherung der Dateisysteme, Verzeichnisse und Dateien gelten. Wenn Sie Ihre Dateisysteme sichern, können Sie Dateien oder Dateisysteme im Falle eines Festplattenfehlers wiederherstellen. Es gibt verschiedene Methoden für das Sichern von Informationen.

#### **Zugehörige Informationen:**

Adding an Unsupported Device to the System

#### **Systemstart für BSD-4.3-Systemmanager:**

Im Folgenden wird der Start des Systems AIX für BSD-4.3-Systemmanager beschrieben.

Auf BSD-4.3-Systemen ist das Programm **init** der letzte Schritt in der Startprozedur. Die Hauptaufgabe des Programms **init** ist das Erstellen von Prozessen für jeden verfügbaren Terminal-Port. Die verfügbaren Terminal-Ports können durch Lesen der Datei `/etc/ttys` ermittelt werden.

In System V wird das Programm **init** bei der Systeminitialisierung gestartet. Der Prozess **init** startet die Prozesse gemäß den Einträgen in der Datei `/etc/inittab`.

AIX folgt der Initialisierungsprozedur von System V. Sie können die Datei `/etc/inittab` direkt, mit dem Befehl **telinit** oder mit den folgenden Befehlen editieren:

Eintrag	Beschreibung
<b>chitab(1)</b>	Ändert Datensätze in der Datei /etc/inittab.
<b>lsitab(1)</b>	Listet Datensätze in der Datei /etc/inittab auf.
<b>mkitab(1)</b>	Erstellt Datensätze in der Datei /etc/inittab.
<b>rmitab(1)</b>	Entfernt Datensätze aus der Datei /etc/inittab.

An der Datei /etc/inittab vorgenommene Änderungen werden beim nächsten Warmstart des Systems bzw. bei der Ausführung des Befehls **telinit q** wirksam.

### Dateien für BSD-4.3-Systemmanager suchen und überprüfen:

Die folgende Liste enthält die BSD-Dateibefehle, die AIX unterstützt.

AIX unterstützt die folgenden BSD-4.3-Dateibefehle:

- **which**
- **whereis**
- **what**
- **file**.

AIX bietet keine Unterstützung für die Syntax für Schnellsuche (**ffind**, **fast find**) des BSD-4.3-Befehls **find**. Derzeit gibt es keine Ersatzfunktion. Das folgende Shell-Script **ffind** kann verwendet werden, um die Funktionalität zu simulieren:

```
#!/bin/bsh
PATH=/bin
for dir in /bin /etc /lib /usr
do
find $dir -print | egrep $1
done
```

Die Syntax für das Script **ffind** ist wie folgt:

```
ffind Dateiname
```

### Paging-Bereich für BSD-4.3-Systemmanager:

Die folgenden Befehle unterstützen Sie bei der Verwaltung des Paging-Bereichs (auch Auslagerungsspeicher genannt).

Eintrag	Beschreibung
<b>chps(1)</b>	Ändert die Attribute eines Paging-Bereichs.
<b>lsps(1)</b>	Listet die Attribute eines Paging-Bereichs auf.
<b>mkps(1)</b>	Fügt dem System zusätzlichen Paging-Bereich hinzu.
<b>rmps(1)</b>	Entfernt Paging-Bereich vom System.
<b>swapoff(1)</b>	Inaktiviert einen oder mehrere Paging-Bereiche.
<b>swapon(1)</b>	Gibt zusätzliche Einheiten für Paging und Auslagerung an.

Wenn ein großer Paging-Bereich benötigt wird, verwenden Sie einen logischen Datenträger auf jeder Festplatte für das Paging. Damit ist das Einrichten von Paging-Bereichen über mehrere Plattenlaufwerke möglich.

### Standardstartprozedur für die Unterstützung der BSD-4.3-ASCII-Konfiguration ändern:

Sie können Netzchnittstellen für AIX mit den SMIT- und ODM-Dateien oder mit den BSD-4.3-ASCII-Konfigurationsdateien verwalten.

Zur Verwaltung der Netzchnittstellen mit BSD-4.3-ASCII-Konfigurationsdateien müssen Sie die Kommentarzeichen für die Befehle in der Datei `/etc/rc.net` unter der folgenden Überschrift entfernen:

```
Part II - Traditional
Configuration
```

Wenn Sie die Konfiguration mit unstrukturierten Dateien und die SRC-Unterstützung verwenden möchten, öffnen Sie die Datei `/etc/rc.net` in einem Editor, und entfernen Sie die Kommentarzeichen für die Befehle **hostname**, **ifconfig** und **route** mit den entsprechenden Parametern.

Wenn Sie die Konfiguration mit unstrukturierten Dateien, aber ohne SRC-Unterstützung verwenden möchten, verwenden Sie den Direktaufruf **smit setbootstrap\_option**, um das System auf die **rc**-Konfiguration im BSD-Stil umzustellen. Diese Option konfiguriert das System so, dass es beim Starten die Datei `/etc/rc.bsdnet` verwendet. Außerdem müssen Sie die Datei `/etc/rc.bsdnet` editieren und die Kommentarzeichen für die Befehle **hostname**, **ifconfig** und **route** mit den entsprechenden Parametern entfernen.

### Zusätzliche Optionen für die Befehle **ifconfig** und **netstat**:

Die folgende Liste enthält zusätzliche Optionen für die Befehle **ifconfig** und **netstat**.

Der Befehl **ifconfig** für AIX hat die folgenden zusätzlichen Optionen:

**mtu** Die Variable *mtu* gibt die größte zu übertragende Einheit (MTU, Maximum Transmission Unit), die im lokalen Netz (oder lokalen Teilnetzen) verwendet wird, und die für ferne Netze verwendete größte zu übertragende Einheit an. Um eine maximale Kompatibilität mit Ethernet und anderen Netzen zu erreichen, setzen Sie den *mtu*-Standardwert für Token-Ring und Ethernet auf 1500.

#### **allcast**

Das Flag **allcast** legt die Broadcast-Strategie für Token-Ring fest. Die Angabe des Flags **allcast** optimiert die Konnektivität über Token-Ring-Brücken. Wenn Sie das Flag **allcast** negieren (mit `-allcast`), wird übermäßiger Datenverkehr im Token-Ring auf ein Minimum reduziert.

Der Befehl **netstat** für AIX hat das Flag `-v`. Der Befehl **netstat -v** gibt Treiberstatistiken aus, wie z. B. die Anzahl übertragener Bytes, die Anzahl der Übertragungsfehler, die Anzahl empfangener Bytes und die Anzahl der Empfangsfehler. Weitere Informationen finden Sie in den Beschreibungen der Befehle **ifconfig** und **netstat** in der Veröffentlichung "Commands Reference".

### Zusätzliche Befehle für die Netzverwaltung:

Die folgenden zusätzlichen Befehle werden in AIX unterstützt.

Eintrag	Beschreibung
<b>securetcpip</b>	Das Shell-Script <b>securetcpip</b> ermöglicht einen kontrollierten Zugriffsmodus, der erweiterte Netzsicherheit bietet. Er unterbindet die Ausführung mehrerer nicht gesicherter TCP/IP-Programme, wie z. B. <b>ftp</b> , <b>rcp</b> , <b>rlogin</b> und <b>rsh</b> . Außerdem schränkt er die Verwendung der Datei <code>.netrc</code> ein.
<b>gated</b>	Der Befehl <b>gated</b> stellt die MIB-Unterstützung für SNMP bereit.

Eintrag	Beschreibung
<b>no</b>	Der Befehl <b>no</b> definiert Netzoptionen, wie z. B.:
	<b>dogticks</b> Definiert die Zeitgeberunterteilung für die Routinen <b>ifwatchdog</b> .
	<b>subnetsarelocal</b> Bestimmt, ob sich die Paketadresse im lokalen Netz befindet.
	<b>ipsendredirects</b> Gibt an, ob der Kernel Umleitungssignale senden soll.
	<b>ipforwarding</b> Gibt an, ob der Kernel Pakete weiterleiten soll.
	<b>tcp_ttl</b> Gibt die Lebensdauer (TTL, Time-to-Live) für TCP-Pakete (Transmission Control Protocol) an.
	<b>udp_ttl</b> Gibt die Lebensdauer (TTL, Time-to-Live) für UDP-Pakete (User Datagram Protocol) an.
	<b>maxttl</b> Gibt die Lebensdauer (TTL, Time-to-Live) für RIP-Pakete (Routing Information Protocol) an.
	<b>ipfragttl</b> Gibt die Lebensdauer (TTL, Time-to-Live) für IP-Fragmente (Internet Protocol) an.
	<b>lowclust</b> Gibt die untere Grenze für den <b>mbuf</b> -Pool des Cluster an.
	<b>lowmbuf</b> Gibt die untere Grenze für den <b>mbuf</b> -Pool an.
	<b>thewall</b> Gibt an, wie viel Speicher dem <b>mbuf</b> -Pool und dem <b>mbuf</b> -Pool des Cluster maximal zugeordnet wird.
	<b>arpt_killc</b> Gibt die Zeit (in Minuten) an, nach der ein inaktiver vollständiger ARP-Eintrag (Address Resolution Protocol) gelöscht wird.
<b>iptrace</b>	Der Befehl <b>iptrace</b> unterstützt die Paketwegverfolgung auf Schnittstellenebene für Internet-Protokolle.
<b>ipreport</b>	Der Befehl <b>ipreport</b> konvertiert den Trace in ein lesbares Format. Im Folgenden sehen Sie ein Beispiel für die Verwendung dieses Befehls: <pre>iptrace -i en0 /tmp/iptrace.log # kill iptrace daemon kill `ps ax   grep iptrace   awk '{ print \$1 }'` ipreport /tmp/iptrace.log   more</pre>

### Eine BSD-4.3-Kennwortdatei importieren:

Sie können eine BSD-4.3-Kennwortdatei in AIX importieren.

Führen Sie zum Importieren einer BSD-4.3-Kennwortdatei die folgenden Schritte aus:

1. Kopieren Sie die BSD-4.3-Kennwortdatei in die Datei `/etc/passwd`, und geben Sie Folgendes ein:  

```
pwdck -y ALL
```
2. Fügen Sie der Datei `/etc/security/limits` für jeden neuen Benutzer eine leere Zeilengruppe hinzu. Der Befehl **usrck** führt diese Aktion ordnungsgemäß aus. Wenn Sie jedoch den Befehl **usrck** verwenden, kann dies zu Problemen führen, sofern die Datei `/etc/group` nicht mit der Datei `/etc/passwd` importiert wurde. Weitere Informationen finden Sie in der Beschreibung des Befehls **usrck** in der Veröffentlichung "Commands Reference".  
**Achtung:** Wenn die Datei `/etc/security/limits` geändert wird, darf der Wert für `stack` 65.536 Bytes nicht überschreiten. Sollte der Wert größer sein, kann es bei der Ausführung des Befehls **usrck** zu Problemen kommen. Ändern Sie den Wert für `stack` in 65.536, und führen Sie den Befehl **usrck** erneut aus.
3. Führen Sie die Befehle **grpck** und **usrck** aus, um die Gruppen- und Benutzerattribute zu prüfen.



## Kennwortdatei für BSD-4.3-Systemmanager bearbeiten:

Im Folgenden wird erläutert, wie Einträge in der Kennwortdatei geändert und wie Kennwörter in AIX nach BSD-4.3-Vorgehensweise verwaltet werden.

In AIX werden die Befehle **lsuser**, **mkuser**, **chuser** und **rmuser** für die Verwaltung von Kennwörtern bereitgestellt. Diese Befehle können mit SMIT ausgeführt werden. Die Ausführung dieser Befehle bezieht sich jedoch jeweils auf einen Benutzer.

Weitere Informationen finden Sie in den Beschreibungen der Befehle **lsuser**, **mkuser**, **chuser** und **rmuser** in der Veröffentlichung "Commands Reference".

**Anmerkung:** Wenn Sie einen Editor verwenden, um mehrere Benutzernameneinträge gleichzeitig zu ändern, müssen Sie mehrere Dateien gleichzeitig bearbeiten, da Kennwörter in der Datei `/etc/security/passwd`, Berechtigungsdaten in der Datei `/etc/security/user` und die verbleibenden Benutzerdaten in der Datei `/etc/passwd` gespeichert sind.

AIX bietet keine Unterstützung für den Befehl **vipw**, unterstützt aber den Befehl **mkpasswd**. Trotzdem können Sie Kennwörter in AIX im BSD-4.3-Stil verwalten. Gehen Sie wie folgt vor:

1. Kopieren Sie eine BSD-4.3-Kennwortdatei in die Datei `/etc/shadow`.
2. Ändern Sie die Berechtigungen für die Datei mit dem folgenden Befehl:  
`chmod 000 /etc/shadow`
3. Kopieren Sie das folgende Shell-Script **vipw** in das Verzeichnis `/etc`:

```


#!/bin/bsh

vipw. Uses pwdck for now. May use usrck someday

PATH=/bin:/usr/bin:/etc:/usr/ucb # Add to this if your editor is
 # some place else
if [-f /etc/ptmp] ; then
 echo "/etc/ptmp exists. Is someone else using vipw?"
 exit 1
fi
if [! -f `which "$EDITOR" | awk '{ print $1 }'`] ; then
 EDITOR=vi
fi
cp /etc/shadow /etc/ptmp
if (cmp /etc/shadow /etc/ptmp) ; then
 $EDITOR /etc/ptmp
else
 echo cannot copy shadow to ptmp
 exit 1
fi
if (egrep "^root:" /etc/ptmp >/dev/null) ; then
 cp /etc/ptmp /etc/shadow ; cp /etc/ptmp /etc/passwd
 chmod 000 /etc/passwd /etc/shadow
 pwdck -y ALL 2>1 >/dev/null # return code 114 may change
 rc=$?
 if [$rc -eq 114] ; then
 chmod 644 /etc/passwd
 rm -f /etc/passwd.dir /etc/passwd.pag
 mkpasswd /etc/passwd
 # update /etc/security/limits, or ftp
 # will fail
 else
 pwdck -y ALL
 fi
else
 echo bad entry for root in ptmp
```

```
fi
rm /etc/ptmp
```

4. Wenn Sie das Shell-Script **vipw** oder den Befehl **mkpasswd** verwenden, beachten Sie, dass SMIT und die Befehle **mkuser**, **chuser** und **rmuser** den Befehl **mkpasswd** nicht verwenden. Sie müssen den Befehl

```
mkpasswd /etc/passwd
```

ausführen, um die Dateien `/etc/passwd.dir` und `/etc/passwd.pag` zu aktualisieren.

**Achtung:** Die Initialisierung der Variablen *IFS* und die trap-Anweisungen dienen als Vorkehrung gegen gebräuchliche Methoden zur Ausnutzung von Sicherheitslücken in der Funktion **setuid**. Die Shell-Scripts **vipw** und **passwd** sind jedoch für relativ offene Umgebungen bestimmt, in denen Kompatibilität ein wichtiger Aspekt ist. Wenn Sie eine Umgebung mit höherer Sicherheit wünschen, verwenden Sie nur die Standardbefehle für AIX.

5. Kopieren Sie das folgende Shell-Script **passwd** in das Verzeichnis `/usr/ucb`:

```

#!/bin/ksh
#
matches changes to /etc/security/passwd file with changes to
#/etc/shadow
#
IFS=" "
PATH=/bin
trap "exit 2" 1 2 3 4 5 6 7 8 10 12 13 14 15 16 17 18 21 22 \
 23 24 25 27 28 29 30 31 32 33 34 35 36 60 61 62
if [-n "$1"]; then
 USERNAME=$1
else
 USERNAME=$LOGNAME
fi
if [-f /etc/ptmp]; then
 echo password file busy
 exit 1
fi
 trap "rm /etc/ptmp; exit 3" 1 2 3 4 5 6 7 8 10 12 13 \
 14 15 16 17 18 21 22 23 24 25 27 28 29 30 31 \
 32 33 34 35 36 60 61 62
if (cp /etc/security/passwd /etc/ptmp) ; then
 chmod 000 /etc/ptmp else
 rm -f /etc/ptmp exit 1
fi
if (/bin/passwd $USERNAME) ; then
 PW=`awk ' BEGIN { RS = "" }
 $1 == user { print $4 } ' user="$USERNAME:" \
/etc/security/passwd `
else
 rm -f /etc/ptmp
 exit 1
fi
rm -f /etc/ptmp
awk -F: '$1 == user { print $1:"pw":'$3 "':"$4":'$5":'$6":'$7 }
 $1 != user { print $0 }' user="$USERNAME" pw="$PW" \
 /etc/shadow > /etc/ptmp
chmod 000 /etc/ptmp
mv -f /etc/ptmp /etc/shadow

```

6. Ändern Sie die Berechtigungen für das Script **passwd** mit dem folgenden Befehl:
- ```
chmod 4711 /usr/ucb/passwd
```
7. Vergewissern Sie sich, dass die Umgebungsvariable *PATH* jedes Benutzers angibt, dass das Verzeichnis `/usr/ucb` vor dem Verzeichnis `/bin` durchsucht wird.

Leistungsmessung und -optimierung für BSD-4.3-Systemmanager

Im Folgenden werden die Einheitenattribute sowie Leistungsmessung und -optimierung in AIX beschrieben.

Allen Einheiten in AIX sind Attribute zugeordnet. Geben Sie Folgendes ein, um Einheitenattribute anzuzeigen:

```
lsattr -E -l Einheitenname
```

Alle Attribute mit dem Wert true können mit dem folgenden Befehl geändert werden:

```
chdev -l Einheitenname -a Attr=Wert
```

Achtung: Wenn Sie beim Ändern der Einheitenparameter Fehler machen, kann dies Ihr System beschädigen.

Standardmäßig ist die maximale Anzahl an Prozessen pro Benutzer 40. Der Standardwert kann für Benutzer, die viele Fenster gleichzeitig geöffnet haben, zu niedrig sein. Mit dem folgenden Befehl kann der Wert systemweit geändert werden:

```
hdev -l sys0 -a maxuproc=100
```

Dieser Beispielbefehl ändert die maximale Anzahl in 100. Der neue Wert wird einmal beim Neustart des Systems gesetzt.

Geben Sie Folgendes ein, um die aktuelle Einstellung dieses und anderer Systemattribute anzuzeigen:

```
lsattr -E -l sys0
```

Das Attribut **maxmbuf** wird derzeit von den mbuf-Services nicht unterstützt.

AIX unterstützt die Befehle **vmstat** und **iostat**, aber weder den Befehl **sysstat** noch Durchschnittslasten. Weitere Informationen finden Sie in den Beschreibungen der Befehle **vmstat** und **iostat** in der Veröffentlichung "Commands Reference".

Drucker für BSD-4.3-Systemmanager

Das Betriebssystem AIX unterstützt das Betriebssystem zwei Druckersubsysteme: BSD 4.3 und System V.

Das Druckersubsystem des Typs System V verwendet Befehle, Warteschlangen und Dateien von System V Release 4 und wird auf dieselbe Weise verwaltet. Die folgenden Abschnitte enthalten die erforderlichen Informationen für die Verwaltung des Druckersubsystems des Typs BSD 4.3. Sie steuern, welches Subsystem über SMIT aktiviert wird. Es kann jeweils nur ein Subsystem aktiviert werden.

Druckoperationen werden von Programmen und Konfigurationen im Verzeichnis `/usr/lpd` verwaltet. Das Design, die Konfiguration, die Warteschlangenmechanismen und die Dämonprozesse der BSD-4.3-Druckersubsysteme für AIX sind verschieden. Beide verwenden jedoch das Protokoll **lpd** für das Drucken über ferne Systeme. Beide Systeme verwenden `/etc/hosts.lpd`, sofern vorhanden, bzw. `/etc/host.equiv`. Das Druckersubsystem für AIX stellt ein Gateway zum Druckersubsystem von BSD 4.3 bereit, so dass Systeme, die AIX verwenden, Druckjobs an BSD-4.3-Systeme übergeben und Druckjobs akzeptieren können, die von BSD-4.3-Systemen übergeben werden.

Die Datei `/etc/printcap` von BSD 4.3 ist in AIX nicht vorhanden. Diese Datei ist eine Kombination aus Spooler-Konfiguration und Datenbasis für die Druckerfunktionen. Benutzer müssen mit dem Format und den Schlüsselwörtern der Datei `printcap` vertraut sein, um einen Drucker ordnungsgemäß konfigurieren zu können.

Die Datei /etc/qconfig von AIX enthält nur die Konfigurationsdaten für den Spooler. Die Druckerfunktionen werden in der vordefinierten bzw. angepassten ODM-Datenbasis definiert. Mit dem Befehl **mkvirprt** (make virtual printer, virtuellen Drucker erstellen) können Sie im System die Funktionen eines bestimmten Druckers definieren.

Geben Sie beispielsweise Folgendes in der Datei /etc/printcap des BSD-4.3-Systems ein, wenn Sie auf dem Drucker lp0 auf dem fernen Host viking drucken möchten:

```
lp0|Print on remote printer attached to
viking:Z
:lp=:rm=viking:rp=lp:st=/usr/spool/lp0d
```

Wenn Sie dasselbe in AIX tun möchten, fügen Sie Folgendes in die Datei /etc/qconfig ein:

```
lp0:
    device = dlp0
    host = viking
    rq = lp
dlp0:
    backend = /usr/lib/lpd/rembak
```

AIX unterstützt die folgenden Druckerbefehle und Bibliotheksfunktionen:

| Eintrag | Beschreibung |
|----------------------|--|
| cancel(1) | Bricht Anforderungen an einen Zeilendrucker ab. |
| chqueuedev(1) | Ändert die Einheitennamen für Drucker oder Plotter. |
| chvirprt(1) | Ändert die Attributwerte eines virtuellen Druckers. |
| disable(1) | Inaktiviert eine Druckerwarteschlange. |
| enable(1) | Aktiviert eine Druckerwarteschlange. |
| hplj(1) | Führt eine Nachbearbeitung einer troff -Ausgabe für einen Drucker des Typs HP LaserJetII mit K-Kassette durch. |
| ibm3812(1) | Führt eine Nachbearbeitung einer troff -Ausgabe für einen Seitendrucker des Typs IBM 3812 Mod 2 durch. |
| ibm3816(1) | Führt eine Nachbearbeitung einer troff -Ausgabe für einen Seitendrucker des Typs IBM 3816 durch. |
| ibm5587G(1) | Führt eine Nachbearbeitung einer troff -Ausgabe für einen Drucker des Typs IBM 5587G mit einer Kassette des Typs 32x32/24x24 durch. |
| lp(1) | Sendet Anforderungen an einen Zeilendrucker. |
| lpr(1) | Reiht Druckjobs in die Warteschlange ein. |
| lprm(1) | Entfernt Jobs aus der Spool-Warteschlange des Zeilendruckers. |
| lpstat(1) | Zeigt Statusinformationen für den Zeilendrucker an. |
| lpctest(1) | Generiert das Testmuster für den Zeilendrucker. |
| lsallqdev(1) | Listet die Einheitennamen aller konfigurierten Druckereinheiten in einer Warteschlange auf. |
| lsvirprt(1) | Zeigt die Attributwerte eines virtuellen Druckers an. |
| mkque(1) | Fügt dem System eine Druckerwarteschlange hinzu. |
| mkqueuedev(1) | Fügt dem System eine Druckerwarteschlangeneinheit hinzu. |
| mkvirprt(1) | Erstellt einen virtuellen Drucker. |
| pac(1) | Bereitet Abrechnungsdatensätze für Drucker/Plotter vor. |
| piobe(1) | Druckjobmanager für das Drucker-Back-End. |
| pioburst(1) | Generiert Trennseiten (Kopf- und Abschlusseiten) für die Druckausgabe. |
| piocmdout(3) | Subroutine, die eine Attributzeichenfolge für das Formatierungsprogramm eines Druckers ausgibt. |
| piodigest(1) | Verarbeitet die Attributwerte für die Definition eines virtuellen Druckers und speichert diese. |
| pioexit(3) | Subroutine, die das Formatierungsprogramm eines Druckers beendet. |
| pioformat(1) | Steuert das Formatierungsprogramm eines Druckers. |
| piofquote(1) | Konvertiert bestimmte Steuerzeichen, die für PostScript-Drucker bestimmt sind. |
| piogetstr(3) | Subroutine, die eine Attributzeichenfolge für das Formatierungsprogramm eines Druckers abrufen. |
| piogetvals(3) | Subroutine, die Variablen für die Druckerattributdatenbank für das Formatierungsprogramm eines Druckers initialisiert. |
| piomsgout(3) | Subroutine, die eine Nachricht vom Formatierungsprogramm eines Druckers sendet. |
| pioout(1) | Schnittstellenprogramm des Einheitentreibers für das Drucker-Back-End. |
| piopredef(1) | Erstellt eine vordefinierte Definition für Druckerdatenströme. |
| proff(1) | Formatiert Text für Drucker mit PC-Druckerdatenströmen. |

| Eintrag | Beschreibung |
|-------------|---|
| qadm(1) | Führt die Systemverwaltung für das Spool-System eines Druckers durch. |
| qconfig(4) | Konfiguriert ein Druckerwarteschlangensystem. |
| qstatus(1) | Stellt den Druckerstatus für das Spool-System für Drucker bereit. |
| restore(3) | Versetzt den Drucker in den Standardzustand. |
| rmque(1) | Entfernt eine Druckerwarteschlange vom System. |
| rmquedev(1) | Entfernt eine Drucker- oder Plotterwarteschlangeneinheit vom System. |
| rmvirprt(1) | Entfernt einen virtuellen Drucker. |
| splp(1) | Ändert oder zeigt die Einstellungen der Druckertreiber an. |
| xpr(1) | Formatiert ein Bildschirmabbild (Screenshot) für die Ausgabe auf einem Drucker. |

Zugehörige Informationen:

Printer Overview for System Management

Systemverwaltungsbefehle für die Systemmanager von BSD 4.3

Diese Liste enthält Befehle, die speziell für die Verwaltung der Umgebung für AIX bestimmt sind.

| Eintrag | Beschreibung |
|--------------|---|
| bosboot(1) | Initialisiert eine Booteinheit. |
| bootlist(1) | Ändert die Liste der für das System verfügbaren Booteinheiten (bzw. die Reihenfolge dieser Einheiten in der Liste). |
| cfgmgr(1) | Konfiguriert Einheiten durch Ausführung der Programme im Verzeichnis /etc/methods. |
| chcons(1) | Leitet die Systemkonsole nach dem nächsten Systemstart an eine Einheit oder in eine Datei um. |
| chdev(1) | Ändert die Merkmale einer Einheit. |
| chdisp(1) | Ändert die vom LFT-Subsystem (Low-Function Terminal) verwendete Anzeige. |
| checkcw(1) | Bereitet Text mit konstanter Breite für den Befehl troff vor. |
| checkeq(1) | Prüft Dokumente, die mit Memorandum-Makros formatiert sind. |
| checkmm(1) | Prüft Dokumente, die mit Memorandum-Makros formatiert sind. |
| checknr(1) | Prüft nroff- und troff-Dateien. |
| chfont(1) | Ändert die zur Bootzeit ausgewählte Standardschriftart. |
| chfs(1) | Ändert die Attribute eines Dateisystems. |
| chgroup(1) | Ändert die Attribute für Gruppen. |
| chgrpmem(1) | Ändert die Administratoren oder Mitglieder einer Gruppe. |
| chhwkbd(1) | Ändert die LFT-Tastaturattribute, die in der ODM-Datenbank (Object Data Manager) gespeichert sind. |
| chitab(1) | Ändert Datensätze in der Datei /etc/inittab. |
| chkbd(1) | Ändert die Standardtastaturbelegung, die vom LFT beim Systemstart verwendet wird. |
| chkey(1) | Ändert den Verschlüsselungsschlüssel. |
| chlang | Setzt die Umgebungsvariable <i>LANG</i> in der Datei /etc/environment für die nächste Anmeldung. |
| chlicense(1) | Es gibt zwei Typen von Benutzerlizenzen: feste und Floating-Lizenzen. Standardmäßig aktiviert sind feste Lizenzen, und die Anzahl der Lizenzen kann mit dem Flag -u geändert werden. Floating-Lizenzen können mit dem Flag -f (und den Einstellungen on und off) aktiviert bzw. inaktiviert werden. |
| chlv(1) | Ändert die Merkmale eines logischen Datenträgers. |
| chnamsv(1) | Ändert die Konfiguration des TCP/IP-basierten Namensservice auf einem Host. |
| chprtsv(1) | Ändert die Konfiguration eines Druckservice auf einer Client- oder Servermaschine. |
| chps(1) | Ändert die Attribute eines Paging-Bereichs. |
| chpv(1) | Ändert die Merkmale eines physischen Datenträgers in einer Datenträgergruppe. |
| chque(1) | Ändert den Namen der Warteschlange. |
| chquedev(1) | Ändert die Einheitenamen für Druck- und Plotterwarteschlangen. |
| chssys(1) | Ändert eine Subsystemdefinition in der Objektklasse subsystem. |
| chtcb(1) | Ändert oder fragt das Attribut für die Computersicherheitsbasis einer Datei ab. |
| chtz | Ändert die Zeitzoneinformationen des Systems. |
| chuser(1) | Ändert Attribute für den angegebenen Benutzer. |
| chvfs(1) | Ändert Einträge in der Datei /etc/vfs. |
| chvg(1) | Definiert die Merkmale einer Datenträgergruppe. |
| chvirprt(1) | Ändert die Attributwerte eines virtuellen Druckers. |
| crfs(1) | Fügt ein Dateisystem hinzu. |
| crvfs(1) | Erstellt Einträge in der Datei /etc/vfs. |

| Eintrag | Beschreibung |
|----------------------|--|
| exportvg(1) | Exportiert die Definition einer Datenträgergruppe aus einer Gruppe physischer Datenträger. |
| extendvg(1) | Fügt einer Datenträgergruppe physische Datenträger hinzu. |
| grpck(1) | Überprüft die Richtigkeit einer Gruppeneffinition. |
| importvg(1) | Importiert eine neue Datenträgergruppeneffinition aus einer Gruppe physischer Datenträger. |
| lsallq(1) | Listet die Namen aller konfigurierten Warteschlangen auf. |
| lsallqdev(1) | Listet die Einheitenamen aller konfigurierten Drucker- und Plottereinheiten in einer bestimmten Warteschlange auf. |
| lsdisp(1) | Listet die derzeit im System verfügbaren Bildschirme auf. |
| lsfont(1) | Listet die für den Bildschirm verfügbaren Schriftarten auf. |
| lsfs(1) | Zeigt die Merkmale von Dateisystemen an. |
| lsgroup(1) | Zeigt die Attribute von Gruppen an. |
| lsitab(1) | Listet die Datensätze in der Datei <code>/etc/inittab</code> auf. |
| lskbd(1) | Listet die derzeit für das LFT-Subsystem verfügbaren Tastaturbelegungen auf. |
| lslicense(1) | Zeigt die Anzahl fester Lizenzen und den Status der Floating-Lizenzen an. |
| lslpp(1) | Listet optionale Programmprodukte auf. |
| lsnamsv(1) | Zeigt die in der Datenbank gespeicherten Informationen zum Namensservice an. |
| lsprtsv(1) | Zeigt die in der Datenbank gespeicherten Informationen zum Druckservice an. |
| lsp | Listet Paging-Bereich und Attribute auf. |
| lsque(1) | Zeigt den Namen der Zeilengruppe für die Warteschlange an. |
| lsquedev(1) | Zeigt den Namen der Einheitenzeilengruppe an. |
| lssrc(1) | Ruft den Status eines Subsystems, einer Subsystemgruppe oder eines Subservers ab. |
| lsuser(1) | Zeigt Attribute von Benutzeraccounts an. |
| lsvfs(1) | Listet Einträge in der Datei <code>/etc/vfs</code> auf. |
| mkcatdefs(1) | Führt eine Vorverarbeitung einer Nachrichtenquellendatei durch. |
| runcat(1) | Leitet die Ausgabedaten des Befehls mkcatdefs über eine Pipe an den Befehl gencat . |
| mkdev(1) | Fügt dem System eine Einheit hinzu. |
| mkfont(1) | Fügt dem System den Schriftartcode hinzu, der einem Bildschirm zugeordnet ist. |
| mkfontdir(1) | Erstellt aus einem Verzeichnis mit Schriftartdateien eine Datei <code>fonts.dir</code> . |
| mkgroup(1) | Erstellt eine neue Gruppe. |
| mkitab(1) | Erstellt Datensätze in der Datei <code>/etc/inittab</code> . |
| mklv(1) | Erstellt einen logischen Datenträger. |
| mklvcopy(1) | Fügt einem logischen Datenträger Kopien hinzu. |
| mknamsv(1) | Konfiguriert den TCP/IP-basierten Namensservice auf einem Host für einen Client. |
| mknotify(1) | Fügt der Objektklasse <code>notify</code> eine <code>notify</code> -Methodendefinition hinzu. |
| mkprtsv(1) | Konfiguriert den TCP/IP-basierten Druckservice auf einem Host. |
| mkps(1) | Fügt dem System zusätzlichen Paging-Bereich hinzu. |
| mkque(1) | Fügt dem System eine Druckerwarteschlange hinzu. |
| mkquedev(1) | Fügt dem System eine Druckerwarteschlangeneinheit hinzu. |
| mkserver(1) | Fügt der Objektklasse <code>subserver</code> eine Subserverdefinition hinzu. |
| mkssys(1) | Fügt der Objektklasse <code>subsystem</code> eine Subsystemdefinition hinzu. |
| mksysb | Sichert angehängte Dateisysteme in der Datenträgergruppe <code>rootvg</code> für eine nachfolgende Neuinstallation. |
| mkszfile | Protokolliert die Größe angehängter Dateisysteme in der Datenträgergruppe <code>rootvg</code> für eine Neuinstallation. |
| mktcpip(1) | Legt die erforderlichen Werte für das Starten von TCP/IP auf einem Host fest. |
| mkuser(1) | Erstellt einen neuen Benutzeraccount. |
| mkuser.sys(1) | Passt einen neuen Benutzeraccount an. |
| mkvg(1) | Erstellt eine Datenträgergruppe. |
| mkvirprt(1) | Erstellt einen virtuellen Drucker. |
| odmadd(1) | Fügt erstellten Objektklassen Objekte hinzu. |
| odmchange(1) | Ändert den Inhalt eines ausgewählten Objekts in der angegebenen Objektklasse. |
| odmcreate(1) | Erzeugt die Dateien <code>.c</code> (Quelle) und <code>.h</code> (Include), die für die Entwicklung von ODM-Anwendungen erforderlich sind, und erstellt leere Objektklassen. |
| odmdelete(1) | Löscht ausgewählte Objekte aus einer angegebenen Objektklasse. |
| odmdrop(1) | Entfernt eine Objektklasse. |
| odmget(1) | Ruft Objekte aus den angegebenen Objektklassen ab und kopiert sie in eine Eingabedatei <code>odmadd</code> . |
| odmshow(1) | Zeigt eine Objektklasseneffinition am Bildschirm an. |
| pwdck(1) | Überprüft die Richtigkeit lokaler Authentifizierungsdaten. |

| Eintrag | Beschreibung |
|---------------------|--|
| redefinevg | Definiert die Gruppe physischer Datenträger der angegebenen Datenträgergruppe in der Einheitenkonfigurationsdatenbank erneut. |
| reducevg(1) | Entfernt physische Datenträger aus einer Datenträgergruppe. Wenn alle physischen Datenträger aus der Datenträgergruppe entfernt wurden, wird die Datenträgergruppe gelöscht. |
| reorgvg(1) | Reorganisiert die Zuordnung physischer Partitionen für eine Datenträgergruppe. |
| restbase(1) | Stellt angepasste Informationen aus dem Boot-Image wieder her. |
| rmDEL(1) | Entfernt ein Delta aus einer SCCS-Datei (Source Code Control System). |
| rmdev(1) | Entfernt eine Einheit aus dem System. |
| rmf(1) | Entfernt Ordner und die darin enthaltenen Nachrichten. |
| rmfs(1) | Entfernt ein Dateisystem. |
| rmgroup(1) | Entfernt eine Gruppe. |
| rmITab(1) | Entfernt Datensätze in der Datei <code>/etc/inittab</code> . |
| rmlv(1) | Entfernt logische Datenträger aus einer Datenträgergruppe. |
| rmlvcopy(1) | Entfernt Kopien von einem logischen Datenträger. |
| rmm(1) | Entfernt Nachrichten. |
| rmnamsv(1) | Dekonfiguriert den TCP/IP-basierten Namensservice auf einem Host. |
| rmnotify(1) | Entfernt eine notify-Methodendefinition aus der Objektklasse notify. |
| rmprtsv(1) | Dekonfiguriert einen Druckservice auf einer Client- oder Servermaschine. |
| rmPS(1) | Entfernt einen Paging-Bereich vom System. |
| rmque(1) | Entfernt eine Druckerwarteschlange vom System. |
| rmqudev(1) | Entfernt eine Warteschlangeneinheit für einen Drucker oder Plotter aus dem System. |
| rmserver(1) | Entfernt eine Subserverdefinition aus der Objektklasse subserver. |
| rmssys(1) | Entfernt eine Subsystemdefinition aus der Objektklasse subsystem. |
| rmuser(1) | Entfernt einen Benutzeraccount. |
| rmvfs(1) | Entfernt Einträge in der Datei <code>/etc/vfs</code> . |
| rmvirprt(1) | Entfernt einen virtuellen Drucker. |
| savebase(1) | Speichert angepasste Basiseinheitendaten in der ODM-Datenbank auf der Booteinheit. |
| swapoff(1) | Inaktiviert mindestens einen Paging-Bereich. |
| swapon(1) | Gibt weitere Einheiten für Paging und Auslagerung (Swapping) an. |
| syncvg(1) | Synchronisiert Kopien logischer Datenträger, die nicht aktuell sind. |
| usrck(1) | Überprüft die Richtigkeit einer Benutzerdefinition. |
| varyoffvg(1) | Inaktiviert eine Datenträgergruppe. |
| varyonvg(1) | Aktiviert eine Datenträgergruppe. |

Zugehörige Konzepte:

„Hauptunterschiede zwischen BSD 4.3 und AIX“ auf Seite 342

Im Folgenden finden Sie eine Übersicht über die Hauptunterschiede zwischen den Systemen AIX und BSD 4.3.

cron für BSD-4.3-Systemmanager

Der Dämon **cron** für dieses Betriebssystem gleicht dem Dämon **cron** von System V Release 2.

Der Dämon **cron** wird mit einem Eintrag in der Datei `/etc/inittab` gestartet.

Einheiten für BSD-4.3-Systemmanager

Im Folgenden sind die Einheiten für BSD-4.3-Systemmanager beschrieben.

Eine Anwendung kann nur dann auf eine Einheit in einem BSD-4.3-System zugreifen, wenn

- die Einheit physisch installiert ist und funktioniert,
- der Treiber für die Einheit im Kernel vorhanden ist,
- die Einheitengeräte-dateien für die Einheit im Verzeichnis `/dev` enthalten sind.

Eine Anwendung kann nur dann auf eine Einheit unter AIX zugreifen, wenn

- die Einheit physisch installiert ist und funktioniert,
- der Treiber für die Einheit im Kernel oder in einer geladenen Kernelerweiterung vorhanden ist,

- die Einheitengeräte-dateien für die Einheit im Verzeichnis /dev enthalten sind,
- die Objektdatenbank im Verzeichnis /etc/objrepos Einträge für die Einheit enthält, die der physischen Konfiguration entsprechen.

Die einheitenspezifischen Programme, die so genannten *Methoden*, die im Verzeichnis /etc/methods enthalten sind, verwalten die Objektdatenbank. Die Methoden werden vom Konfigurationsmanager (mit dem Befehl **cfgmgr**) und von anderen Befehlen aufgerufen.

Wenn ein Anwendungsprogramm nicht mehr auf eine Einheit zugreifen kann, kann dies bedeuten, dass die Hardware fehlerhaft ist oder dass die Konfigurationsdatenbank im Verzeichnis /etc/objrepos beschädigt ist.

Der Befehl **cfgmgr** verarbeitet die Konfigurationsdatenbank im Verzeichnis **/etc/objrepos** und wird zur Startzeit vom Befehl **cfgmgr** (Konfigurationsmanager) verarbeitet.

Der folgende Pseudocode zeigt die Logik des Konfigurationsmanagers:

```

/* Main */
While - Solange Regeln in der Datenbank Config_Rules enthalten sind
{
    Nächste Regel abrufen und ausführen
    stdout der letzten Ausführung erfassen
    Parse_Output(stdout)
}
/* Routine für Ausgabeanalyse (Parse_Output)*/
/* stdout enthält eine Liste der gefundenen Einheiten */
Parse_OutPut(stdout)
{
    While - solange noch Einheiten in der Liste enthalten sind
    {
        Einheit in der Datenbank suchen
        if (!defined)
            Definitionsmethode aus der Datenbank abrufen und ausführen
        if (! configured)
            {
                Konfigurationsmethode aus der Datenbank abrufen und ausführen
                Parse_Output(stdout)
            }
    }
}

```

UUCP für BSD-4.3-Systemmanager

In der folgenden Tabelle sind die UUCP-Befehle und -Dateien aufgelistet.

| Eintrag | Beschreibung |
|-------------------|---|
| Dialers(4) | Listet Modems auf, die für ferne BNU-Kommunikationsverbindungen verwendet werden. |
| Maxuuxqts(4) | Begrenzt die Anzahl der BNU-Dämonprozesse des Typs uuxqt , die ausgeführt werden können. |
| Permissions(4) | Gibt die BNU-Befehlsberechtigungen für ferne Systeme an. |
| Poll(4) | Gibt an, wann das Programm BNU ferne Systeme abfragen soll. |
| Systems(4) | Listet ferne Computer auf, mit denen das lokale System kommunizieren kann. |
| rmail(1) | Bearbeitet ferne Mails, die über BNU empfangen werden. |
| uucheck(1) | Sucht nach Dateien und Verzeichnissen, die von BNU benötigt werden. |
| uuclean(1) | Entfernt Dateien aus dem BNU-Spool-Verzeichnis. |
| uucleanup(1) | Löscht ausgewählte Dateien aus dem BNU-Spool-Verzeichnis. |
| uucpadmin(1) | Gibt grundlegende BNU-Konfigurationsdaten an. |
| uudemon.admin(1) | Liefert in regelmäßigen Abständen Informationen zum Status von Dateiübertragungen mit BNU. |
| uudemon.cleanu(1) | Bereinigt die BNU-Spool-Verzeichnisse und -Protokolldateien. |
| uudemon.hour(1) | Leitet Dateitransportaufrufe an ferne Systeme mit dem Programm BNU ein. |
| uudemon.poll(1) | Frägt die in der BNU-Abfragedatei aufgelisteten Systeme ab. |
| uulog(1) | Stellt Informationen über Dateiübertragungsaktivitäten mit BNU auf einem System bereit. |

| Eintrag | Beschreibung |
|------------------|--|
| uupoll(1) | Erzwingt das Abfragen eines fernen BNU-Systems. |
| uuq(1) | Zeigt die BNU-Jobwarteschlange an und löscht angegeben Jobs aus der Warteschlange. |
| uusnap(1) | Zeigt den Status von BNU-Kontakten mit fernen Systemen an. |
| uustat(1) | Meldet den Status von BNU-Operationen zurück und bietet eine eingeschränkte Steuerung von BNU-Operationen. |

AIX unterstützt auch die BSD-4.3-Befehle **uuencode** und **uudecode**. Der HDB-Befehl **uugetty** wird nicht unterstützt. Weitere Informationen finden Sie in den Beschreibungen der Befehle **uuencode** und **uudecode** in der Veröffentlichung "Commands Reference".

Zugehörige Informationen:

BNU file and directory structure

Dateisysteme für BSD-4.3-Systemmanager

Für das Anhängen (Mount) und Abhängen (Unmount) von Dateisystemen werden ähnliche Befehle verwendet.

AIX verwendet die Datei `/etc/filesystem`, um Informationen zu Dateisystemeinheiten aufzulisten, und hat ähnliche Befehle für das Anhängen und Abhängen von Dateisystemen.

Datei `/etc/filesystems` und Datei `/etc/fstab`:

Auf BSD-4.3-Systemen werden Listen mit Blockeinheiten und Mountpunkten in der Datei `/etc/fstab` gespeichert. Auf SVR4-Systemen werden Informationen zu Blockeinheiten und Mountpunkten in der Datei `/etc/vfstab` gespeichert. In AIX werden Informationen zu Blockeinheiten und Mountpunkten in der Datei `/etc/filesystems` gespeichert.

Die Befehle **crfs**, **chfs** und **rmfs** aktualisieren die Datei `/etc/filesystems`.

Von besonderem Interesse für BSD-4.3-Systemadministratoren kann die Variable *check* in der Datei `/etc/filesystems` sein. Die Variable *check* kann auf True, False oder eine Zahl gesetzt werden. Sie können beispielsweise *check=2* in der Datei `/etc/filesystems` angeben. Die Zahl gibt die Instanz des Befehls **fsck** an, in der dieses Dateisystem geprüft wird. Der Parameter *check* entspricht dem fünften Feld in einem Datensatz der Datei `/etc/fstab`.

In der Datei `/etc/filesystems` gibt es keinen Parameter für die Häufigkeit, mit der Sicherungen erstellt werden.

Dateisystemunterstützung in AIX:

AIX unterstützt verschiedene Dateisysteme.

AIX unterstützt Datenträgerkontingente.

In AIX können Disketten nicht als Dateisysteme angehängt werden.

Die Syntax der Befehle **mount** und **umount** für AIX weicht von den BSD-4.3- und SVR4-Versionen dieser Befehle ab. Die Befehle zum gleichzeitigen Anhängen und Abhängen aller Dateisysteme für alle drei Systeme sind in der folgenden Tabelle aufgeführt:

Befehle **mount** und **umount**

| Funktion | Syntax für dieses Betriebssystem | Syntax für BSD 4.3 | Syntax für SVR4 |
|----------------------------|----------------------------------|--------------------|------------------|
| Alle Dateisysteme anhängen | mount all | mount -a | mountall |
| Alle Dateisysteme abhängen | umount all | umount -a | umountall |

Weitere Informationen finden Sie in „Dateisysteme“ auf Seite 453.

Terminals für **BSD-4.3-Systemmanager**

Im Folgenden werden Terminals für BSD-4.3-Systemmanager beschrieben.

Traditionell aktivieren und inaktivieren BSD-4.3-Systemmanager Terminalanschlüsse, indem sie die Datei `/etc/ttys` ändern und ein Signal des Typs **HUP** an das Programm **init** senden.

AIX speichert Informationen zu Terminalanschlüssen im ODM und startet Terminals, wenn das Programm **init** die Datei `/etc/inittab` liest. In AIX können Sie zum Konfigurieren von Terminalanschlüssen die Anwendung "Einheiten" der Schnittstelle SMIT verwenden.

Es gibt keine feste Zuordnung zwischen dem Anschluss und dem Namen der Einheitengerätefile im Verzeichnis `/dev`. Deshalb ist es für Systemmanager, die mit AIX nicht vertraut sind, nicht eindeutig, welcher Anschluss konfiguriert werden muss. Wenn Sie SMIT verwenden, wird der erste serielle Anschluss auf der Platine (physisch gekennzeichnet mit **s1**) in den SMIT-Menüs als Position **00-00-S1**, Adapter **sa0** und Anschluss **s1** bezeichnet. Der zweite serielle Anschluss auf der Platine (physisch mit **s2** gekennzeichnet) wird als Position **00-00-S2**, Adapter **sa1** und Anschluss **s2** bezeichnet.

Verwenden Sie die Befehle **penable** und **pdisable**, um einen Anschluss zu aktivieren bzw. zu inaktivieren.

termcap und **terminfo**:

Wie System V verwendet dieses System terminfo-Einträge in den Dateien `/usr/lib/terminfo/?./*`.

Auch Benutzer von BSD-4.3-Systemen können die folgenden Befehle hilfreich finden:

captoinfo(1)

Konvertiert eine Datei **termcap** in eine Datei **terminfo**.

tic(1) Übersetzt die **terminfo**-Dateien aus dem Quellenformat in das kompilierte Format.

Dieses Betriebssystem enthält Quellen für viele **terminfo**-Einträge. Einige dieser Einträge müssen möglicherweise mit dem Befehl **tic** kompiliert werden. Die Datei **termcap** wird in der Datei `/lib/libtermcap/termcap.src` bereitgestellt.

Eingabe- und Ausgabeumleitung

Im Betriebssystem AIX können Sie die Eingabe und Ausgabe (E/A) von Daten mit bestimmten E/A-Befehlen und -Symbolen steuern.

Sie können die Eingabe steuern, indem Sie angeben, wo die Daten erfasst werden sollen. Beispielsweise können Sie festlegen, dass auf der Tastatur eingegebene Daten (Standardeingabe) oder Daten aus einer Datei gelesen werden sollen. Sie können die Ausgabe steuern, indem Sie angeben, wo die Daten angezeigt oder gespeichert werden sollen. Beispielsweise können Sie festlegen, dass die Daten auf dem Bildschirm ausgegeben (Standardausgabe) oder in eine Datei geschrieben werden sollen.

Da das Betriebssystem AIX Multitasking unterstützt, ist es so konzipiert, dass Prozesse kombiniert mit anderen ausgeführt werden können.

Zugehörige Konzepte:

„Dateien verwalten“ auf Seite 201

Es gibt zahlreiche Verwendungsmöglichkeiten für die Dateien in Ihrem System. Gewöhnlich wird eine Textdatei mit einem Texteditor erstellt.

„Befehle für das Anzeigen von Dateiinhalten (Befehle `pg`, `more`, `page` und `cat`)“ auf Seite 205

Mit den Befehlen **pg**, **more** und **page** können Sie den Inhalt einer Datei anzeigen und die Geschwindigkeit steuern, mit der die Dateien angezeigt werden.

„Eingabe- und Ausgabeumleitung in der Korn- bzw. POSIX-Shell“ auf Seite 243

Bevor die Korn-Shell einen Befehl ausführt, sucht sie in der Befehlszeile nach Umleitungszeichen. Mit diesen Sonderzeichen wird die Shell angewiesen, die Eingabe und Ausgabe umzuleiten.

Standardeingabe-, Standardausgabe- und Standardfehlerdateien

Beim einem Befehlsstart wird in der Regel erwartet, dass die folgenden Dateien geöffnet sind: Standardeingabe, Standardausgabe und Standardfehler (auch *Fehlerausgabe* oder *Diagnoseausgabe*).

Jeder dieser Dateien ist eine Nummer, ein so genannter *Dateideskriptor* zugeordnet. Beispiel:

| Eintrag | Beschreibung |
|-------------------|----------------------------------|
| Dateideskriptor 0 | Standardeingabe |
| Dateideskriptor 1 | Standardausgabe |
| Dateideskriptor 2 | Standardfehlerausgabe (Diagnose) |

Im Allgemeinen erbt ein Kindprozess diese Dateien von seinem Elternprozess. Alle drei Dateien sind standardmäßig der Workstation zugeordnet (0 der Tastatur, 1 und 2 dem Bildschirm). In der Shell können sie jedoch umgeleitet werden, bevor die Steuerung an einen Befehl übergeben wird.

Wenn Sie einen Befehl ohne Dateinamen eingeben, ist Ihre Tastatur die *Standardeingabe* (oder *stdin*). Nach Abschluss der Befehlsausführung werden die Ergebnisse am Bildschirm angezeigt.

Ihr Bildschirm ist die *Standardausgabe* (oder *stdout*). Standardmäßig übernehmen Befehle die Daten von der Standardeingabe und senden die Ergebnisse an die Standardausgabe.

Fehlernachrichten werden an die Standardfehlerausgabe (oder *stderr*) weitergeleitet. Standardmäßig ist dies der Bildschirm.

Die Standardaktionen für Eingabe und Ausgabe können geändert werden. Beispielsweise können Sie eine Datei als Eingabe verwenden und die Ergebnisse eines Befehls in eine Datei schreiben. Dieses Verfahren wird als *Eingabe-/Ausgabeumleitung* bezeichnet.

Die Ausgabe eines Befehls, die normalerweise an den Bildschirm gesendet wird, kann stattdessen in eine Datei umgeleitet werden. Dieser Vorgang wird als *Ausgabeumleitung* bezeichnet. Die Verwendung der Ausgabeumleitung empfiehlt sich, wenn die Ausgabe umfangreich und am Bildschirm nur schwer zu lesen ist oder wenn Dateien zu einer größeren Datei zusammengefasst werden sollen.

Die Umleitung wird bei der Eingabe zwar nicht so häufig verwendet wie bei der Ausgabe, es ist jedoch generell möglich, die Eingabe für einen Befehl, die normalerweise über die Tastatur erfolgt, von einer Datei umzuleiten. Dieser Vorgang wird als *Eingabeumleitung* bezeichnet. Mit der Eingabeumleitung können Dateien vorbereitet und anschließend vom Befehl gelesen werden.

Standardausgabeumleitung

Wenn Sie am Ende eines Befehls die Angabe *>Dateiname* hinzufügen, wird die Ausgabe des Befehls in die angegebene Datei geschrieben. Das Symbol *>* wird auch als *Operator für die Ausgabeumleitung* bezeichnet.

Die Ausgabe aller Befehle, die ihre Ergebnisse normalerweise am Bildschirm ausgeben, kann in eine Datei umgeleitet werden.

Ausgabe in eine Datei umleiten

Sie können die Ausgabe eines Prozesses in eine Datei umleiten, indem Sie den Befehl gefolgt vom Operator für Ausgabeumleitung und einem Dateinamen eingeben.

Wenn Sie beispielsweise die Ergebnisse des Befehls **who** in eine Datei mit dem Namen `users` umleiten möchten, geben Sie Folgendes ein:

```
who > users
```

Anmerkung: Wenn die Datei `users` bereits vorhanden ist, wird sie gelöscht und ersetzt, sofern die Option **noclobber** des integrierten **ksh**- (Korn-Shell) oder **csch**-Befehls (C-Shell) **set** nicht angegeben ist.

Geben Sie zum Anzeigen des Inhalts der Datei `users` Folgendes ein:

```
cat users
```

Es wird eine Liste ähnlich der folgenden angezeigt:

```
denise 1ft/0 May 13 08:05
marta pts/1 May 13 08:10
endrica pts/2 May 13 09:33
```

Ausgabe umleiten und an eine Datei anfügen

Wenn die Notation `>> Dateiname` am Ende eines Befehls hinzugefügt wird, wird die Ausgabe des Befehls an die angegebene Datei angefügt und nicht über vorhandene Daten geschrieben. Das Symbol `>>` wird auch als *Operator für die Umleiten mit Anfügen* bezeichnet.

Wenn Sie beispielsweise Datei2 an Datei1 anfügen möchten, geben Sie Folgendes ein:

```
cat Datei2 >> Datei1
```

Anmerkung: Wenn die Datei `Datei1` nicht vorhanden ist, wird sie erstellt, sofern die Option **noclobber** des integrierten **ksh**-Befehls (Korn-Shell) bzw. **csch**-Befehls (C-Shell) **set** nicht angegeben ist.

Textdatei durch Umleitung von der Tastatur erstellen

Wenn Sie den Befehl **cat** ohne Optionen eingeben, verwendet er alles, was Sie auf der Tastatur eingeben. Diese Eingabe kann in eine Datei umgeleitet werden.

Setzen Sie den Cursor in eine neue Zeile, und drücken Sie anschließend die Tastenkombination Strg-D, um das Textende zu kennzeichnen.

Geben Sie an der Eingabeaufforderung folgenden Befehl ein:

```
cat > Dateiname
Dies ist ein Test.
^D
```

Verknüpfung von Textdateien

Sie können mehrere Dateien zu einer Datei zusammenführen. Die Zusammenführung mehrerer Dateien zu einer Datei wird als *Verknüpfung* bezeichnet.

Das folgende Beispiel erstellt die Datei4, die sich aus den Dateien Datei1, Datei2 und Datei3 zusammensetzt, die in der folgenden Reihenfolge aneinander angefügt werden.

Beispiele:

- Geben Sie an der Eingabeaufforderung folgenden Befehl ein:

```
cat Datei1 Datei2 Datei3 > Datei4
```
- Im Folgenden wird ein typischer Fehler bei der Verknüpfung von Dateien gezeigt:

```
cat Datei1 Datei2 Datei3 > Datei1
```

Achtung: In diesem Beispiel könnten Sie erwarten, dass der Befehl `cat` den Inhalt von `Datei1`, `Datei2` und `Datei3` in `Datei1` anfügt. Der Befehl `cat` erstellt jedoch zuerst die Ausgabedatei, d. h., er löscht den Inhalt von `Datei1` und fügt dann `Datei2` und `Datei3` an diese Datei an.

Standardeingabeumleitung

Wenn Sie am Ende eines Befehls die Angabe `< Dateiname` hinzufügen, wird die Eingabe des Befehls aus der angegebenen Datei gelesen. Das Symbol `<` wird auch als *Operator für die Eingabeumleitung* bezeichnet.

Anmerkung: Nur Befehle, die ihre Eingabe normalerweise über die Tastatur erhalten, können ihre Eingabe umleiten.

Wenn Sie beispielsweise die Datei `Brief1` mit dem Befehl `mail` als Nachricht an Benutzer `denise` senden möchten, geben Sie Folgendes ein:

```
mail denise < Brief1
```

Ausgabe mit der Datei `/dev/null` verwerfen

Die Datei `/dev/null` ist eine Gerätedatei. Diese Datei hat eine eindeutige Eigenschaft: Sie ist immer leer. Alle an die Datei `/dev/null` gesendeten Daten werden verworfen. Diese Funktion ist hilfreich, wenn Sie einen Befehl oder ein Programm ausführen, dessen Ausgabe ignoriert werden soll.

Beispiel: Das Programm `meinprog` akzeptiert Eingaben vom Bildschirm und generiert während der Ausführung Nachrichten, die eigentlich nicht am Bildschirm angezeigt werden sollen. Geben Sie Folgendes ein, um die Eingabe aus der Datei `meinscript` zu lesen und die Standardausgabenachrichten zu verwerfen:

```
meinprog < meinscript >/dev/null
```

In diesem Beispiel verwendet das Programm `meinprog` die Datei `meinscript` als Eingabe und verwirft die gesamte Standardausgabe.

Standardfehler- und andere Ausgaben umleiten

Zusätzlich zur Standardeingabe und Standardausgabe erzeugen Befehle oft andere Arten von Ausgaben, wie beispielsweise Fehler- oder Statusnachrichten, die auch als Diagnosenachrichten bezeichnet werden. Wie die Standardausgabe wird die Standardfehlerausgabe am Bildschirm ausgegeben, sofern sie nicht umgeleitet wird.

Wenn Sie die Standardfehlerausgabe oder andere Ausgaben umleiten möchten, verwenden Sie einen Dateideskriptor. Bei einem *Dateideskriptor* handelt es sich um eine Nummer, die jeder Ein-/Ausgabedatei, die von einem Befehl verwendet wird, zugeordnet wird. Dateideskriptoren können auch zum Umleiten der Standardeingabe und -ausgabe verwendet werden. Die folgenden Nummern sind Standardeingabe, Standardausgabe und Standardfehlerausgabe zugeordnet:

| Eintrag | Beschreibung |
|---------|------------------------------|
| 0 | Standardeingabe (Tastatur) |
| 1 | Standardausgabe (Bildschirm) |
| 2 | Standardfehler (Bildschirm) |

Wenn Sie die Standardfehlerausgabe umleiten möchten, geben Sie vor den Umleitungssymbolen für Ausgabe oder Anfügen (`>` oder `>>`) die Nummer des Dateideskriptors (2) und hinter dem Symbol einen Dateinamen ein. Im folgenden Beispiel wird mit dem Befehl `cc` zunächst die Datei `testfile.c` kompiliert und anschließend die Standardfehlerausgabe dieses Befehls in der Datei `ERRORS` angefügt:

```
cc testfile.c 2 >> ERRORS
```

Mit den Dateideskriptoren 0 bis 9 können auch andere Arten von Ausgaben umgeleitet werden. Wenn der Befehl `cmd` beispielsweise seine Ausgabe in den Dateideskriptor 9 schreibt, können Sie diese Ausgabe mit dem folgenden Befehl in die Datei `savedata` umleiten:

```
cmd 9> savedata
```

Wenn ein Befehl in mehrere Ausgaben schreibt, können diese unabhängig voneinander umgeleitet werden. Beispiel: Ein Befehl leitet seine Standardausgabe an Dateideskriptor 1, die Fehlerausgabe an Dateideskriptor 2 und erstellt eine Datendatei mit Dateideskriptor 9. Die folgende Befehlszeile leitet jede dieser Ausgaben in eine andere Datei um:

```
Befehl > standard 2> error 9> data
```

Ausgabe an Inline-Eingabedokumente (Here) umleiten

Sie können Ausgaben an Inline-Eingabedokumente (Here) umleiten.

Wenn Sie einen Befehl im Format

```
Befehl << Dateiendezeichenfolge
```

eingeben und *Dateiendezeichenfolge* eine Zeichenfolge ist, die keine Platzhalterzeichen enthält, interpretiert die Shell alle folgenden Zeilen als Standardeingabe für den *Befehl*, bis sie eine Zeile liest, die lediglich *eofstring* (unter Umständen mit einem oder mehreren vorangestellten Tabulatorzeichen) enthält. Die Zeilen zwischen dem ersten und dem zweiten *eofstring* werden häufig als *Inline-Eingabedokument* oder *Here-Dokument* bezeichnet. Wenn direkt hinter den Umleitungszeichen << ein Minuszeichen (-) steht, schneidet die Shell die führenden Tabulatorzeichen der einzelnen Zeilen des **Here**-Dokuments ab, bevor sie die Zeilen an *Befehl* übergibt.

Die Shell erstellt eine temporäre Datei mit dem Inhalt des **Here**-Dokuments und führt dort die Variablen- und Befehlssubstitution durch, bevor sie die Datei an den Befehl übergibt. Sie führt eine Mustererkennung für die Dateinamen durch, die Teil von Befehlszeilen in Befehlssubstitutionen sind. Wenn Sie Substitutionen vollständig verhindern möchten, setzen Sie ein Zeichen von *eofstring* in Anführungszeichen.

```
Befehl << \eofstring
```

Das **Here-Dokument** empfiehlt sich besonders für geringe Eingabedatenmengen, die sich eher für die Shellprozedur als eine separate Datei (wie Editorscripts) eignen. Sie könnten beispielsweise Folgendes angeben:

```
cat <<- xyz
  Diese Nachricht wird ohne führende
  Tabulatorzeichen angezeigt.
xyz
```

Zugehörige Konzepte:

„Eingabe- und Ausgabeumleitung in der Korn- bzw. POSIX-Shell“ auf Seite 243

Bevor die Korn-Shell einen Befehl ausführt, sucht sie in der Befehlszeile nach Umleitungszeichen. Mit diesen Sonderzeichen wird die Shell angewiesen, die Eingabe und Ausgabe umzuleiten.

Ausgabe mit Pipes und Filtern umleiten

Sie können mehrere Befehle so miteinander verbinden, dass die Standardausgabe eines Befehls als Standardeingabe für einen anderen Befehl verwendet wird. Auf diese Art miteinander verbundene Befehle werden als *Befehlskette* bezeichnet.

Die Verbindung zwischen diesen Befehlen wird als *Pipe* bezeichnet. Pipes sind hilfreich, weil sie viele Einzelbefehle zu einem leistungsstarken Befehl zusammenführen können. Mit einer Befehlskette können Sie die Ausgabe eines Befehls als Eingabe für einen anderen Befehl weiterleiten. Die Befehle werden mit einem Pipe-Symbol (|) verbunden.

Wenn ein Befehl seine Eingabe von einem anderen Befehl erhält, diese ändert und seine Ergebnisse an die Standardausgabe sendet, wird dies als *Filter* bezeichnet. Filter können eigenständig verwendet werden, empfehlen sich aber insbesondere in Befehlsketten. Die am häufigsten verwendeten Filter sind im Folgenden aufgeführt:

- sort

- more
- pg

Beispiele:

- Der Befehl **ls** schreibt den Inhalt des aktuellen Verzeichnisses in einen fortlaufenden Datenstrom, der am Bildschirm angezeigt wird. Wird hierfür mehr als eine Bildschirmseite benötigt, gehen einige Daten in der Ansicht verloren. Damit die Ausgabe seitenweise am Bildschirm angezeigt wird, können Sie eine Befehlskette verwenden, die die Ausgabe des Befehls **ls** an den Befehl **pg** leitet, der das Ausgabeformat für den Bildschirm steuert. Geben Sie beispielsweise Folgendes ein:

```
ls | pg
```

In diesem Beispiel wird die Ausgabe des Befehls **ls** als Eingabe für den Befehl **pg** übernommen. Drücken Sie die Eingabetaste, um die nächste Bildschirmseite anzuzeigen.

Befehlsketten arbeiten nur in einer Richtung (von links nach rechts). Jeder Befehl in einer Befehlskette wird als separater Prozess ausgeführt, aber alle Prozesse können parallel ausgeführt werden. Ein Prozess wird angehalten, wenn er keine Eingabe erhält oder der Eingabepuffer für den Folgeprozess voll ist.

- Ein weiteres Beispiel für die Verwendung von Pipes ist der Befehl **grep**. Der Befehl **grep** durchsucht eine Datei nach Zeilen, die Zeichenfolgen eines bestimmten Musters enthalten. Wenn Sie beispielsweise alle Dateien anzeigen möchten, die im Juli erstellt oder geändert wurden, geben Sie Folgendes ein:

```
ls -l | grep Jul
```

In diesem Beispiel wird die Ausgabe des Befehls **ls** als Eingabe für den Befehl **grep** übernommen.

Programmausgabe anzeigen und in eine Datei kopieren (Befehl tee)

Wenn der Befehl **tee** mit einer Pipe verwendet wird, liest er die Standardeingabe, schreibt anschließend die Ausgabe eines Programms in die Standardausgabe und kopiert sie parallel in die angegebenen Dateien. Mit dem Befehl **tee** können Sie die Ausgabe sofort anzeigen und gleichzeitig zur späteren Verwendung speichern.

Geben Sie beispielsweise Folgendes ein:

```
ps -ef | tee program.ps
```

Dieser Befehl zeigt die Standardausgabe des Befehls **ps -ef** am Bildschirm an und speichert gleichzeitig eine Kopie dieser Ausgabe in der Datei `program.ps`. Falls die Datei `program.ps` bereits vorhanden ist, wird sie gelöscht und neu angelegt, sofern die Option **noclobber** des integrierten Befehls **set** nicht definiert ist.

Geben Sie beispielsweise Folgendes ein, um die Ausgabe eines Befehls anzuzeigen und in einer bereits vorhandenen Datei zu speichern:

```
ls -l | tee -a program.ls
```

Dieser Befehl zeigt die Standardausgabe von **ls -l** am Bildschirm an und fügt gleichzeitig eine Kopie dieser Ausgabe am Ende der Datei `program.ls` an.

Das System zeigt Informationen ähnlich den folgenden an. Die Datei `program.ls` enthält dieselben Informationen:

```
-rw-rw-rw- 1 jones  staff  2301  Sep 19  08:53 161414
-rw-rw-rw- 1 jones  staff  6317  Aug 31  13:17 def.rpt
-rw-rw-rw- 1 jones  staff  5550  Sep 10  14:13 try.doc
```

Die vollständige Syntax des Befehls **tee** ist in der Veröffentlichung *Commands Reference, Volume 5* beschrieben.

Bildschirminhalt löschen (Befehl clear)

Mit dem Befehl **clear** können Sie Nachrichten und Tastatureingaben vom Bildschirm löschen.

Geben Sie an der Eingabeaufforderung Folgendes ein:

```
clear
```

Der Bildschirminhalt wird gelöscht, und die Eingabeaufforderung wird wieder angezeigt.

Nachricht an die Standardausgabe senden

Mit dem Befehl **echo** können Sie Nachrichten am Bildschirm anzeigen.

Geben Sie Folgendes an der Eingabeaufforderung ein, um eine Nachricht in die Standardausgabe zu schreiben:

```
echo Bitte Diskette einlegen . . .
```

Die folgende Nachricht wird angezeigt:

```
Bitte Diskette einlegen . . .
```

Wenn Sie beispielsweise den Befehl **echo** mit Platzhalterzeichen verwenden möchten, geben Sie an der Eingabeaufforderung Folgendes ein:

```
echo Die Sicherungsdateien heißen: *.bak
```

Das System zeigt die Nachricht Die Sicherungsdateien heißen:, gefolgt von den Dateinamen im aktuellen Verzeichnis an, die die Erweiterung .bak haben.

Einzelne Textzeilen in einer Datei anfügen (Befehl echo)

Mit dem Befehl **echo** und dem Anfügesymbol können einzelne Textzeilen in einer Datei angefügt werden.

Geben Sie an der Eingabeaufforderung beispielsweise Folgendes ein:

```
echo Am Ende der Woche daran denken, die Mail-Dateien zu sichern.>>notes
```

Dieser Befehl fügt die Nachricht Am Ende der Woche daran denken, die Mail-Dateien zu sichern. am Ende der Datei notes hinzu.

Bildschirminhalt in eine Datei kopieren (Befehle capture und script)

Mit dem Befehl **capture**, der ein VT100-Terminal emuliert, können Sie alle Daten, die auf Ihrem Terminal ausgegeben werden, in eine beliebige Datei kopieren. Mit dem Befehl **script** können Sie alle Daten, die an Ihrem Terminal ausgegeben werden, in eine beliebige Datei kopieren, ohne ein VT100-Terminal zu emulieren.

Beide Befehle bieten sich für die Ausgabe von Datensätzen von Terminaldialogen an.

Geben Sie Folgendes an der Eingabeaufforderung ein, um den Bildschirminhalt eines Terminals mit VT100-Emulation zu erfassen:

```
capture screen.01
```

Das System zeigt Informationen ähnlich den folgenden an:

```
Befehl capture gestartet. Die Datei  
ist screen.01.
```

```
Mit ^P Dump der Anzeige in Datei screen.01 stellen.
```

```
Emulation eines Terminals vt100 läuft.
```

```
Weiter mit beliebiger Taste.
```

Nach der Eingabe von Daten und dem Speichern der Anzeige in einer Datei können Sie den Befehl **capture** mit der Tastenkombination Strg-D oder mit dem Befehl **exit** stoppen. Das System zeigt Informationen ähnlich den folgenden an:

Befehl `capture` abgeschlossen.
Die Datei ist `screen.01`.
Emulation eines Terminals `vt100` ist NICHT MEHR aktiv.

Mit dem Befehl `cat` kann der Inhalt der Datei angezeigt werden.

Geben Sie Folgendes an der Eingabeaufforderung ein, um den Bildschirminhalt eines Terminals ohne VT100-Emulation zu erfassen:

```
script
```

Das System zeigt Informationen ähnlich den folgenden an:

Befehl `script` wurde gestartet. Die Datei ist `typescript`.

Die gesamte Anzeige am Bildschirm wird jetzt in die Datei `typescript` kopiert.

Zum Abbrechen des Befehls `script` drücken Sie die Tastenkombination Strg-D, oder geben Sie den Befehl `exit` ein. Das System zeigt Informationen ähnlich den folgenden an:

Die Ausführung des Befehls `script` ist beendet. Die Datei ist `typescript`.

Mit dem Befehl `cat` kann der Inhalt der Datei angezeigt werden.

Die vollständige Syntax können Sie den Beschreibungen der Befehle `capture` und `script` in der Veröffentlichung *Commands Reference* entnehmen.

Text in Großbuchstaben am Bildschirm anzeigen (Befehl `banner`)

Mit dem Befehl `banner` werden ASCII-Zeichen in Großschrift am Bildschirm angezeigt.

Jede Zeile der Ausgabe kann bis zu 10 Zeichen (Groß- oder Kleinbuchstaben) enthalten.

Geben Sie an der Eingabeaufforderung beispielsweise Folgendes ein:

```
banner AUF WIEDERSEHEN!
```

Das System zeigt `AUF WIEDERSEHEN!` in Großbuchstaben am Bildschirm an.

Befehlsübersicht für Eingabe- und Ausgabeumleitung

Im Folgenden finden Sie eine Übersicht über die Befehle für Eingabe- und Ausgabeumleitung.

| Eintrag | Beschreibung |
|------------------------|---|
| <code>></code> | „Standardausgabeumleitung“ auf Seite 363 |
| <code><</code> | „Standardeingabeumleitung“ auf Seite 365 |
| <code>> ></code> | „Ausgabe umleiten und an eine Datei anfügen“ auf Seite 364 |
| <code> </code> | „Ausgabe mit Pipes und Filtern umleiten“ auf Seite 366 |
| <code>banner</code> | Gibt ASCII-Zeichenfolgen in Großbuchstaben in der Standardausgabe aus. |
| <code>capture</code> | Gibt den Inhalt der Terminalanzeige in einer Datei aus. |
| <code>clear</code> | Löscht die Terminalanzeige. |
| <code>echo</code> | Gibt Zeichenfolgen in die Standardausgabe aus. |
| <code>script</code> | Kopiert die Terminalein- und ausgaben in eine Datei. |
| <code>tee</code> | Zeigt die Standardausgabe eines Programms an und kopiert sie in eine Datei. |

AIX-Kernelwiederherstellung

Beginnend mit AIX 6.1 kann der Kernel optional nach Fehlern in ausgewählten Routinen wiederhergestellt werden, so dass ein unvorhergesehener Systemausfall vermieden werden kann.

Die Kernelwiederherstellung ist standardmäßig inaktiviert. Wenn die Kernelwiederherstellung aktiviert ist, kann das System kurzzeitig während der Ausführung einer Kernelwiederherstellungsaktion angehal-

ten werden. Dieser Zeitraum ist im Allgemeinen kleiner als zwei Sekunden. Unmittelbar nach der Wiederherstellungsaktion für den Kernel finden folgende Aktionen statt:

- Die Systemkonsole zeigt die folgende Nachricht an:

```
-----  
Es wurde eine Wiederherstellungsaktion nach einem Kernelfehler ausgeführt. Ein Wiederherstellungsprotokoll wurde  
-----
```

- AIX fügt einen Eintrag in das Fehlerprotokoll ein. Sie können die Fehlerprotokolldaten zu Servicezwecken an IBM senden, ähnlich wie Sie die Daten einer vollständigen Systembeendigung senden. Es folgt ein Beispiel für einen Fehlerprotokolleintrag bei Ausführung der Wiederherstellungsaktion:

```
LABEL:          RECOVERY  
Date/Time:      Fri Feb 16 14:04:17 CST 2007  
Type:          INFO  
Resource Name:  RMGR  
Beschreibung  
Kernel Recovery Action  
Detaildaten  
Live Dump Base Name  
RECOV_20070216200417_0000  
Function Name  
w_clear  
FRR Name  
w_init_clear_frr  
Symptom String  
273  
EEEE00009627A072  
F10001001B18BBC0  
w_clear+D0  
wdog0030+288  
test_index+4C  
Recovery Log Data  
0001 0000 0000 0000 F000 0000 2FFC AEB0 0000 0111 0000 0000 0000 0000 0021 25BC  
8000 0000 0002 9032 EEEE 0000 9627 A072 F100 0100 1B18 BBC0 0000 0000 0000 0000  
0000 0001 0000 0000 0006 0057 D2FF 8C00 0001 0148 0500 0000 8000 0000 0002 9032  
.....
```

- AIX generiert einen zeitnahen Speicherauszug. Die Daten aus dem zeitnahen Speicherauszug sind standardmäßig im Verzeichnis `/var/adm/ras/livedump` enthalten, und die Datei erhält den Namen **RECOV_Zeitmarke_Zahl**, wobei *Zeitmarke* den Zeitpunkt angibt, zu dem die Kernelwiederherstellung erfolgte, und *Zahl* die Anzahl der Kernelwiederherstellungsaufrufe. Sie können die zeitnahen Speicherauszugsdaten zu Servicezwecken an IBM senden, ähnlich wie Sie die Daten einer vollständigen Systembeendigung senden. Nähere Informationen zu zeitnahen Speicherauszügen finden Sie unter live dumps in der Veröffentlichung *Kernel Extensions and Device Support Programming Concepts*.

Achtung: Nach einer Kernelwiederherstellung sind einige Funktionen möglicherweise nicht mehr verfügbar. Das Betriebssystem bleibt jedoch in einem stabilen Zustand. Führen Sie gegebenenfalls einen Systemabschluss mit anschließendem Neustart durch, um die fehlenden Funktionen wiederherzustellen.

Hinweise zu Speicher und Prozessor

AIX verwaltet während der Kernelhauptoperationen Daten zum Status der Kernelwiederherstellung. Wenn die Kernelwiederherstellung aktiviert wird, sind zusätzliche Prozessoranweisungen zum Verwalten der Daten und zusätzlicher Speicher zum Speichern der Daten erforderlich. Die Auswirkungen auf die Prozessorauslastung sind minimal. Die zusätzliche Speicherbelegung kann mithilfe der folgenden Gleichung berechnet werden, wobei *maxThread* die maximale Anzahl der auf dem System ausgeführten Threads ist und *Prozessoranzahl* die Anzahl der Prozessoren.

Erforderlicher Speicher = 4 KB x *maxThread* + 128 KB x *Prozessoranzahl*

Wie im folgenden Beispiel gezeigt, belegt ein System mit 16 Prozessoren und maximal 1000 Threads einen zusätzlichen Speicher von 6304 KB:

$4 \times 1000 + 128 \times 16 = 6304 \text{ KB}$

Kernelwiederherstellung aktivieren und inaktivieren

Sie können die Kernelwiederherstellung über die SMIT-Pfadschnittstelle aktivieren oder inaktivieren:

Verwenden Sie den folgenden SMIT-Pfad, um die Kernelwiederherstellung zu aktivieren oder zu inaktivieren:

Fehlerbestimmung > Kernelwiederherstellung > Status der Kernelwiederherstellung ändern > Status der Kernelwiederherstellung für nächsten Bootvorgang ändern

Verwenden Sie den folgenden SMIT-Pfad, um den aktuellen Status der Kernelwiederherstellung anzuzeigen:

Fehlerbestimmung > Kernelwiederherstellung > Status der Kernelwiederherstellung anzeigen

Einheitenmanagement

Sie können Befehle für die Verwaltung der verschiedenen Einheiten verwenden, die in AIX verfügbar sind. Zu den Einheiten, die Sie verwalten können, gehören Logical Volume Manager, Dateisysteme, Bandlaufwerke und Drucker.

Logical Volume Manager

Die Gruppe von Betriebssystembefehlen, Bibliothekssubroutinen und weiteren Tools, mit denen Sie die Speicherung logischer Datenträger konfigurieren und steuern können, wird als Logical Volume Manager (LVM) bezeichnet.

Der LVM steuert Plattenressourcen, indem er eine Datenzuordnung zwischen einer einfacheren und flexibleren *logischen* Sicht des Speicherbereichs und den echten *physischen* Platten herstellt. Der LVM verwendet hierfür eine Schicht mit Einheitentreibercode, der den traditionellen Platteneinheitentriibern übergeordnet ausgeführt wird.

Der LVM setzt sich aus dem Einheitentreiber für logische Datenträger (LVDD, Logical Volume Device Driver) und der Schnittstellenbibliothek für LVM-Subroutinen zusammen. Der LVDD ist ein Pseudoeinheitentreiber, der alle E/As verwaltet und verarbeitet. Er übersetzt logische Adressen in physische Adressen und sendet E/A-Anforderungen an bestimmte Einheitentreiber. Die *Schnittstellenbibliothek für LVM-Subroutinen* enthält Routinen, die von den Systemverwaltungsbefehlen verwendet werden, um Systemverwaltungstasks für die logischen und physischen Datenträger eines Systems auszuführen.

Zugehörige Informationen:

Logical Volume Programming Overview

Understanding the Logical Volume Device Driver

LVM-Konzepte

Bevor Sie mit der Verwendung von LVM (Logical Volume Manager) beginnen, müssen Sie sich mit den grundlegenden Mechanismen und der grundlegenden Terminologie vertraut machen.

Datenträgergruppe aktivieren:

Das Aktivieren einer Datenträgergruppe (Vary-On) ist einer der Mechanismen, den der LVM verwendet, um sicherzustellen, dass eine Datenträgergruppe zur Verwendung bereit ist und aktuelle Daten enthält.

Die Befehle **varyonvg** und **varyoffvg** aktivieren bzw. inaktivieren eine auf dem System definierte Datenträgergruppe, d. h., sie machen sie verfügbar bzw. nicht verfügbar. Die Datenträgergruppe muss aktiviert werden, bevor das System auf sie zugreifen kann. Während der Aktivierung der Datenträgergruppe liest der LVM Verwaltungsdaten von den physischen Datenträgern, die in der Datenträgergruppe definiert sind. Diese Verwaltungsdaten, zu denen ein Deskriptorbereich der Datenträgergruppe (VGDA, Volume

Group Descriptor Area) und ein Statusbereich der Datenträgergruppe (VGSA, Volume Group Status Area) gehören, sind auf jedem physischen Datenträger der Datenträgergruppe gespeichert.

Der VGDA enthält Informationen, die die Zuordnung physischer Partitionen zu logischen Partitionen für jeden logischen Datenträger in der Datenträgergruppe beschreiben, sowie weitere elementare Daten, einschließlich einer Zeitmarke. Der VGSA enthält Informationen, z. B. welche physischen Partitionen veraltet sind und welche physischen Datenträger fehlen (d. h. nicht verfügbar oder nicht aktiv sind), wenn versucht wird, eine Datenträgergruppe zu aktivieren.

Wenn bei der Aktivierung der Datenträgergruppe auf einen oder mehrere physische Datenträger, die in der Datenträgergruppe definiert sind, nicht zugegriffen werden kann, zeigt der Befehl die Namen aller physischen Datenträger, die für diese Datenträgergruppe definiert sind, und ihren Status an. Diese Informationen helfen Ihnen zu entscheiden, ob die Datenträgergruppe inaktiviert (Vary-Off) werden muss.

Zugehörige Konzepte:

„Hohe Verfügbarkeit bei einem Plattenausfall“ auf Seite 428

Zu den wichtigsten Mitteln für den Schutz gegen Plattenausfälle gehören die Konfigurationseinstellungen für logische Datenträger, wie z. B. Spiegelung.

Quorum:

Das Quorum ist einer der Mechanismen, den der LVM verwendet, um sicherzustellen, dass eine Datenträgergruppe zur Verwendung bereit ist und aktuelle Daten enthält.

Ein Quorum ist ein Votum über die Anzahl der aktiven Deskriptorbereiche der Datenträgergruppe (VGDA, Volume Group Descriptor Areas) und der aktiven Statusbereiche der Datenträgergruppe (VGSA, Volume Group Status Areas). Ein Quorum gewährleistet die Datenintegrität der VGDA's und VGSA's bei einem Plattenausfall. Jede physische Platte in einer Datenträgergruppe hat mindestens einen VGDA und einen VGSA. Wenn eine Datenträgergruppe auf einer einzelnen Platte erstellt wird, hat sie zunächst zwei VGDA's und VGSA's auf der Platte. Enthält eine Datenträgergruppe zwei Platten, hat eine Platte weiterhin zwei VGDA's und VGSA's, aber die andere Platte nur einen VGDA und VGSA. Wenn die Datenträgergruppe drei oder mehr Platten enthält, wird jeder Platte nur ein VGDA und VGSA zugeordnet.

Ein Quorum ist verloren, wenn mindestens die Hälfte der Platten (d. h. ihre VGDA/VGSA-Bereiche) von LVM nicht gelesen werden können. Wenn in einer Datenträgergruppe mit zwei Platten die Platte mit nur einem VGDA und einem VGSA verloren geht, ist das Quorum noch vorhanden, weil trotzdem noch zwei der drei VGDA's und VGSA's erreichbar sind. Wenn die Platte mit zwei VGDA's und VGSA's verloren geht, gilt dies nicht mehr. Je mehr Platten in einer Datenträgergruppe enthalten sind, desto geringer ist das Risiko, dass das Quorum verloren geht, wenn eine Platte ausfällt.

Wenn das Quorum verloren geht, inaktiviert sich die Datenträgergruppe selbst, so dass die Platten für den LVM nicht mehr zugänglich sind. Dies verhindert weitere Platten-E/As in dieser Datenträgergruppe, damit keine Daten verloren gehen oder als geschrieben angenommen werden, wenn physische Probleme auftreten. Nach dem Inaktivieren der Datenträgergruppe wird der Benutzer im Fehlerprotokoll darüber informiert, dass ein Hardwarefehler aufgetreten ist und Wartungsarbeiten durchgeführt werden müssen.

Es gibt Fälle, in denen die weitere Verwendung der Datenträgergruppe wünschenswert ist, selbst wenn ein Quorum verloren geht. In diesen Fällen kann die Quorumprüfung für die Datenträgergruppe inaktiviert werden. Dieser Typ von Datenträgergruppe wird als *Datenträgergruppe ohne Quorum* bezeichnet. Datenträgergruppen ohne Quorum werden am häufigsten verwendet, wenn logische Datenträger gespiegelt wurden. Wenn eine Platte verloren geht, gehen die Daten nicht verloren, wenn sich eine Kopie des logischen Datenträgers auf einer Platte befindet, die nicht inaktiviert und damit zugänglich ist. Es können jedoch Situationen in Datenträgergruppen ohne Quorum (gespiegelt oder nicht gespiegelt) auftreten, in denen sich die Daten (einschließlich der Kopien) auf den nicht verfügbaren Platten befinden. In diesen Fällen sind die Daten unter Umständen nicht zugänglich, selbst wenn die Datenträgergruppe aktiviert bleibt.

Zugehörige Konzepte:

„Eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum konvertieren“ auf Seite 375
Sie können eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum ändern, wenn Sie eine fortlaufende Verfügbarkeit der Daten gewährleisten möchten, auch wenn kein Quorum erreicht wird.

Spiegelpools:

Mithilfe von Spiegelpools können die physischen Datenträger einer Datenträgergruppe auf verschiedene Pools verteilt werden.

Ein Spiegelpool setzt sich aus einem oder mehreren physischen Datenträgern zusammen. Jeder physische Datenträger kann jeweils nur zu einem einzigen Spiegelpool gehören. Wenn Sie einen logischen Datenträger erstellen, können Sie jede erstellte Kopie dieses logischen Datenträgers einem Spiegelpool zuordnen. Den Kopien logischer Datenträger, die einem Spiegelpool zugeordnet sind, werden nur Partitionen der physischen Datenträger zugeordnet, die sich in demselben Spiegelpool befinden. Auf diese Weise können die Platten beschränkt werden, die von einer Kopie eines logischen Datenträgers verwendet werden können. Wenn keine Spiegelpools verwendet werden, ist eine Zuordnungsdatei die einzige Möglichkeit, die physischen Datenträger zu beschränken, die beim Erstellen oder Erweitern eines logischen Datenträgers für die Zuordnung verwendet werden können. Die Verwendung von Spiegelpools vereinfacht diesen Prozess somit erheblich. Spiegelpools können mit dem Befehl **extendvg** oder dem Befehl **chpv** erstellt werden.

Wenn Sie einen neuen Spiegelpool erstellen, müssen Sie einen Namen für den Spiegelpool angeben. Die Namen von Spiegelpools müssen den folgenden Regeln entsprechen.

- Die Namen dürfen nur alphanumerische Zeichen, Unterstreichungszeichen (), Minuszeichen (-) oder Punkte (.) enthalten.
- Sie dürfen maximal 15 Zeichen enthalten.
- Sie müssen in der Datenträgergruppe eindeutig sein.

Sobald Spiegelpools in einer Datenträgergruppe verwendet werden, kann die Datenträgergruppe nicht mehr in eine Version von AIX importiert werden, die keine Spiegelpools unterstützt. Dazu gehören alle Versionen von AIX vor Version 6.1.1.0. Außerdem müssen alle Knoten im Cluster Spiegelpools unterstützen, um Spiegelpools im erweiterten Parallelverarbeitungsmodus verwenden zu können.

Strenge des Spiegelpools

Durch Einstellung der Strenge des Spiegelpools können strengere Einschränkungen für die Verwendung des Speicherpools erzwungen werden. Die folgenden drei Werte können für die Strenge des Speicherpools festgelegt werden:

- off** Wenn Sie den Wert **off** als Einstellung für die Strenge des Spiegelpools angeben, gelten keine Einschränkungen für die Verwendung des Spiegelpools. Dies ist der Standardwert.
- on** Wenn Sie den Wert **on** als Einstellung für die Strenge des Spiegelpools angeben, muss jede Kopie eines logischen Datenträgers, die in der Datenträgergruppe erstellt wird, einem Spiegelpool zugeordnet werden.
- super** Wenn Sie den Wert **super** als Einstellung für die Strenge des Spiegelpools angeben, gelten die folgenden Einschränkungen:
 - Lokale und ferne physische Datenträger dürfen nicht zu demselben Spiegelpool gehören.

Anmerkung: Weitere Informationen zu lokalen und fernen physischen Datenträgern finden Sie in der Dokumentation zu HACMP/XD GLVM.

- In einer Datenträgergruppe kann es maximal drei Spiegelpools geben.
- Jeder Spiegelpool muss mindestens eine Kopie jedes logischen Datenträgers in der Datenträgergruppe enthalten.

Geographic Logical Volume Manager:

Mit Geographic Logical Volume Manager (GLVM) können Sie eine Spiegelkopie Ihrer Daten an einem geografisch fernen Standort verwalten.

GLVM kann Ihr Geschäft durch die Spiegelung kritischer Daten an einem fernen Standort für Wiederherstellung nach einem Katastrophenfall vor einer Katastrophe bewahren. Wenn ein Katastrophenfall eintritt, z. B. Feuer oder Hochwasser, und Ihre Daten am Produktionsstandort zerstört werden, haben Sie durch die Spiegelung eine Sicherungskopie der Daten von Ihrem Standort für Wiederherstellung nach einem Katastrophenfall.

Die Daten werden über TCP/IP-Standardnetze gespiegelt. Der Produktionsstandort und der Standort für Wiederherstellung nach einem Katastrophenfall müssen sich nicht in demselben physischen Netz befinden. Der Einsatz von Routern und Gateways zwischen den beiden Standorten ist zulässig. An Stelle extrem langer Plattenkabel werden das TCP/IP-Netz und der RPV-Einheitentreiber (Remote Physical Volume, Ferner physischer Datenträger) für den Zugriff auf die ferne Platte verwendet.

Der Benutzer konfiguriert die Platten, die geografisch voneinander entfernt sind, als ferne physische Datenträger und kombiniert diese fernen physischen Datenträger anschließend mit lokalen physischen Datenträgern zu geografisch gespiegelten Datenträgergruppen. Diese Datenträgergruppen werden von Logical Volume Manager (LVM) verwaltet und funktionieren ähnlich wie Standarddatenträgergruppen. GLVM unterstützt synchrones und asynchrones Spiegeln über Fernzugriff.

Datenträgergruppen ohne Quorum:

Der Logical Volume Manager (LVM) inaktiviert die Datenträgergruppe automatisch, wenn ihm eine Mehrheit von Deskriptorbereichen der Datenträgergruppe (VGDA, Volume Group Descriptor Areas) oder Statusbereichen der Datenträgergruppe (VGSA, Volume Group Status Areas) fehlt. Sie können jedoch eine Option auswählen, die der Gruppe ermöglicht, so lange online zu bleiben, wie ein VGDA/VGSA-Paar intakt ist. Diese Option erzeugt eine *Datenträgergruppe ohne Quorum*.

Der LVM muss auf alle Platten in Datenträgergruppen ohne Quorum zugreifen können, bevor er die Reaktivierung zulässt. Auf diese Weise wird sichergestellt, dass VGDA und VGSA auf dem aktuellen Stand sind.

Sie können eine Datenträgergruppe ohne Quorum in Systemen erstellen, in denen jeder logische Datenträger mindestens zwei Kopien hat. Wenn eine Platte ausfällt, bleibt die Datenträgergruppe so lange aktiv, wie noch eine aktive Platte vorhanden ist.

Anmerkung: Benutzerdefinierte Datenträgergruppen und die Datenträgergruppe **rootvg** können ohne Quorum verwendet werden, aber die Methoden für die Konfiguration benutzerdefinierter Datenträgergruppen und **rootvg** als Datenträgergruppen ohne Quorum und für die Wiederherstellung nach Hardwarefehlern sind unterschiedlich. Verwenden Sie stets die korrekte Methode für die entsprechende Datenträgergruppe.

Selbst wenn Sie Datenträgergruppen ohne Quorum verwenden, *kann* das Quorum verloren gehen und die folgende Nachricht in der Ausgabe des Befehls **errpt** angezeigt werden.

```
QUORUM LOST, VOLUME GROUP CLOSING LVM.
```

Diese Nachricht wird ausgegeben, wenn alle physischen Datenträger den Status `missing` (fehlt) haben und der LVM die Datenträgergruppe automatisch inaktiviert.

Die Angabe `QUORUM LOST` (Quorumverlust) ist in der Nachricht enthalten, weil durch das Inaktivieren des Quorums für eine Datenträgergruppe die Quorumanforderung auf 1 herabgesetzt wird. Mit dem Befehl **lsvg *Datenträgergruppenname*** können Sie den Quorumwert anzeigen, der im Feld `QUORUM:` ausgewiesen

wird. Sollten alle physischen Datenträger den Status `missing` haben, wird selbst diese Mindestquorumanforderung nicht eingehalten, was zu der oben genannten Nachricht und zum automatischen Inaktivieren der Datenträgergruppe führt.

Zugehörige Informationen:

↳ Logical Volume Manager from A to Z: Introduction and Concepts

Eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum konvertieren:

Sie können eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum ändern, wenn Sie eine fortlaufende Verfügbarkeit der Daten gewährleisten möchten, auch wenn kein Quorum erreicht wird.

Diese Prozedur wird häufig für Systeme mit den folgenden Konfigurationen verwendet:

- Datenträgergruppe mit zwei Platten, in der die logischen Datenträger gespiegelt sind
- Datenträgergruppe mit drei Platten, in der die logischen Datenträger einmal oder zweimal gespiegelt sind

Wenn eine Datenträgergruppe unter diesen Umständen ohne Quorum betrieben werden kann, bleibt sie selbst nach dem Ausfall einer Platte aktiv, solange mindestens eine Platte in der Datenträgergruppe aktiv ist.

Stellen Sie Folgendes sicher, um die Wiederherstellung von Datenträgergruppen ohne Quorum zu gewährleisten:

- Wenn Ihr System JFS- oder JFS2-Dateisysteme verwendet, spiegeln Sie den logischen Datenträger mit dem JFS-Protokoll.
- Speichern Sie gespiegelte Kopien auf separaten Platten. Wenn Sie die Konfiguration nicht genau kennen, geben Sie den folgenden Befehl ein, um die physische Position (PV1, PV2 und PV3) jeder logischen Partition zu prüfen. (Wenn Sie die Kopien auf separaten Platten speichern möchten, müssen die Spalten PV1, PV2 und PV3 unterschiedliche `hdisk`-Nummern enthalten.)

```
lslv -m Name_des_logischen_Datenträgers
```

Wenn die einzigen Kopien eines logischen Datenträgers auf derselben Platte gespeichert sind und diese Platte nicht mehr verfügbar ist, steht auch der Datenträger dem Benutzer nicht mehr zur Verfügung, unabhängig davon, ob es sich um eine Datenträgergruppe mit oder ohne Quorum handelt.

Benutzerdefinierte Datenträgergruppen und die Datenträgergruppe `rootvg` können ohne Quorum verwendet werden, aber ihre Konfigurations- und Wiederherstellungsmethoden sind unterschiedlich.

Zum Aktivieren einer benutzerdefinierten Datenträgergruppe ohne Quorum, müssen alle physischen Datenträger der Datenträgergruppe zugänglich sein, oder die Aktivierung scheitert. Da Datenträgergruppen ohne Quorum so lange online bleiben, bis die letzte Platte ausfällt, muss jede Platte während der Aktivierung zugänglich sein.

Achtung: Wenn eine Platte, die der Datenträgergruppe `rootvg` zugeordnet ist, fehlt, schalten Sie das System nur dann ein, wenn eine Möglichkeit besteht, dass die fehlende Platte repariert werden kann. Der Logical Volume Manager (LVM) verwendet immer das Flag `-f`, um das Aktivieren (Vary-On) der Datenträgergruppe `rootvg` ohne Quorum zu erzwingen. Diese Operation ist jedoch mit Risiken verbunden. Der LVM muss die Aktivierung erzwingen, weil das Betriebssystem nicht gestartet werden kann, wenn `rootvg` nicht aktiviert ist. Anders ausgedrückt, der LVM unternimmt einen letzten Versuch, `rootvg` ohne Quorum zu aktivieren, selbst wenn nur eine einzige Platte zugänglich ist.

Zugehörige Konzepte:

„Hohe Verfügbarkeit bei einem Plattenausfall“ auf Seite 428

Zu den wichtigsten Mitteln für den Schutz gegen Plattenausfälle gehören die Konfigurationseinstellungen für logische Datenträger, wie z. B. Spiegelung.

„Hohe Verfügbarkeit bei einem Adapter- oder Netzteilausfall“ auf Seite 429

Führen Sie eine oder mehrere der im Folgenden beschriebenen Aktionen aus, um sich je nach Anforderungen vor einem Ausfall eines Adapters oder Netzteils zu schützen.

„Richtlinie für Datenträgergruppen implementieren“ auf Seite 442

Nachdem Sie entschieden haben, welche Richtlinien für die Datenträgergruppen Sie verwenden möchten, analysieren Sie Ihre aktuelle Konfiguration, indem Sie in der Befehlszeile den Befehl **lspv** eingeben.

„Quorum“ auf Seite 372

Das Quorum ist einer der Mechanismen, den der LVM verwendet, um sicherzustellen, dass eine Datenträgergruppe zur Verwendung bereit ist und aktuelle Daten enthält.

Logical Volume Manager konfigurieren

Der Logical Volume Manager (LVM) wird zusammen mit dem Basisbetriebssystem installiert und erfordert keine weitere Konfiguration. Es müssen jedoch Platten konfiguriert und als physische Datenträger definiert werden, damit der LVM sie verwenden kann.

Zugehörige Tasks:

„Einen logischen Datenträger ohne Dateisystem für eine Anwendung definieren“ auf Seite 421

Ein *logischer Datenträger ohne Dateisystem* ist ein Bereich physischen und logischen Plattenspeicherplatzes, der direkt von einer Anwendung wie einer Datenbank oder einer Partition und nicht vom Betriebssystem oder einem Dateisystem gesteuert wird.

Verwaltungsbefehle und -direktaufrufe für den LVM:

In der folgenden Tabelle sind die einfachsten Tasks gruppiert, die Sie möglicherweise für die Verwaltung der vom LVM gesteuerten Entitäten (physische und logische Datenträger, Datenträgergruppen und Dateisysteme) benötigen.

Tabelle 62. Tasks für die Verwaltung von logischen Datenträgern und Speicher

| Task | SMIT-Direktaufruf | Befehl oder Datei |
|---|--|---|
| Datenträgergruppe aktivieren | smit varyonvg | |
| Festplatte ohne Daten zu einer vorhandenen Datenträgergruppe hinzufügen | smit extendvg | |
| Festplatte ohne Daten zu einer neuen Datenträgergruppe hinzufügen | smit mkvg | |
| Logischen Datenträger hinzufügen ^{Anmerkung 1} | smit mklv | |
| Datenträgergruppe hinzufügen | smit mkvg | |
| Neue Datenträgergruppe hinzufügen und aktivieren | smit mkvg | |
| Logischen Datenträger auf Datenzuordnung einstellen | smit chlvs | |
| Namen einer Datenträgergruppe ändern ^{Anmerkung 2} | 1. smit varyoffvg
2. smit exportvg
3. smit importvg
4. smit mountfs | 1. varyoffvg <i>AlterDatenträgergruppenname</i>
2. exportvg <i>AlterDatenträgergruppenname</i>
3. importvg <i>NeuerDatenträgergruppenname</i>
4. mount all |
| Datenträgergruppe auf automatische Aktivierung einstellen | smit chvg | |
| Richtlinien für logische Datenträger ändern oder festlegen | smit chlvs | |
| Logischen Datenträger auf einen neuen logischen Datenträger kopieren ^{Anmerkung 3} | smit cplv | |
| Logischen Datenträger auf einen vorhandenen logischen Datenträger derselben Größe kopieren ^{Achtung 1} | smit cplv | |

Tabelle 62. Tasks für die Verwaltung von logischen Datenträgern und Speicher (Forts.)

| Task | SMIT-Direktaufruf | Befehl oder Datei |
|---|---|--|
| Logischen Datenträger auf einen vorhandenen kleineren logischen Datenträger kopieren ^{Achtung 1 Anmerkung 3} | SMIT nicht verwenden ^{Achtung 2} | <ol style="list-style-type: none"> 1. Logischen Datenträger erstellen, z. B. mklv -y hdiskN vg00 4 2. Neues Dateisystem auf einem neuen logischen Datenträger erstellen, z. B. crfs -v jfs -d hdiskN -m /doc -A yes 3. Dateisystem anhängen, z. B. mount /doc 4. Verzeichnis an einem neuen Mountpunkt erstellen, z. B. mkdir /doc/options 5. Dateisystem von einem logischen Quellen- auf einen logischen Zieldatenträger übertragen, z. B. cp -R /usr/adam/oldoptions/* \ /doc/options |
| Logischen Datenträger auf einen vorhandenen größeren logischen Datenträger kopieren ^{Achtung 1} | smit cplv | |
| Datenträgergruppe inaktivieren | smit varyoffvg | |
| Schreiben-Überprüfen (write-verify) aktivieren und Planungsrichtlinie ändern | smit chlvs | |
| Maximale Größe eines logischen Datenträgers erhöhen | smit chlvs | |
| Größe eines logischen Datenträgers erhöhen | smit extendlv | |
| Alle logischen Datenträger nach Datenträgergruppe auflisten | smit lslv2 | |
| Alle physischen Datenträger im System auflisten | smit lspv2 | |
| Alle Datenträgergruppen auflisten | smit lsvg2 | |
| Status, logische Datenträger oder Partitionen auf einem physischen Datenträger auflisten | smit lspv | |
| Inhalt einer Datenträgergruppe auflisten | smit lsvg1 | |
| Status oder Zuordnung eines logischen Datenträgers auflisten | smit lslv | |
| Logischen Datenträger mit oder ohne Datenzuordnung spiegeln | smit mklvcopy | |
| Austauschbaren Datenträger ausschalten | smit offdsk | Nur verfügbar für die Funktion für das Entfernen von Einheiten im laufenden Betrieb |
| Austauschbaren Datenträger einschalten | smit ondsk | Nur verfügbar für die Funktion für das Entfernen von Einheiten im laufenden Betrieb |
| Spiegelung für eine Datenträgergruppe entfernen | smit unmirrorvg | |
| Datenträgergruppe entfernen | smit reducevg2 | |
| Datenträgergruppe reorganisieren | smit reorgvg | |
| Platte dekonfigurieren und ausschalten | smit rmvdsk1 oder smit rmvdsk , anschließend smit opendoor | |

Achtung:

1. Wenn Sie diese Prozedur zum Kopieren auf einen vorhandenen logischen Datenträger verwenden, werden alle Daten auf diesem Datenträger ohne Anforderung einer Benutzerbestätigung überschrieben.
2. Verwenden Sie nicht die SMIT-Prozedur oder den Befehl `cplv`, um einen größeren logischen Datenträger auf einen kleineren zu kopieren. Dies würde zu einem beschädigten Dateisystem führen, weil einige Daten (einschließlich des Superblocks) nicht auf den kleineren logischen Datenträger kopiert werden.

Anmerkung:

1. Nachdem Sie einen logischen Datenträger erstellt haben, hat dieser den Status "Geschlossen" (closed), weil keine LVM-Struktur diesen logischen Datenträger verwendet. Er bleibt geschlossen, bis ein Dateisystem auf dem logischen Datenträger erstellt oder der logische Datenträger für unformatierte Ein-/Ausgabe geöffnet wird.
2. Es ist nicht möglich, die Datenträgergruppe `rootvg` zu importieren, zu exportieren und ihren Namen zu ändern.
3. Sie müssen ausreichend Direktzugriffsspeicher haben, um einen bestimmten logischen Datenträger duplizieren zu können.

Zugehörige Tasks:

„Einen logischen Datenträger ohne Dateisystem für eine Anwendung definieren“ auf Seite 421

Ein *logischer Datenträger ohne Dateisystem* ist ein Bereich physischen und logischen Plattenspeicherplatzes, der direkt von einer Anwendung wie einer Datenbank oder einer Partition und nicht vom Betriebssystem oder einem Dateisystem gesteuert wird.

Platten hinzufügen, während das System verfügbar bleibt:

Die folgende Prozedur beschreibt, wie Sie eine Platte mit der Funktion für das Entfernen von Einheiten im laufenden Betrieb aktivieren und konfigurieren. Diese Funktion ermöglicht Ihnen, Platten hinzuzufügen, ohne das System auszuschalten.

Sie können eine Platte hinzufügen, um zusätzlichen Speicher hinzuzufügen oder um auf einen Plattenausfall zu reagieren. Diese Funktion ist nur auf bestimmten Systemen verfügbar.

1. Installieren Sie die Platte in einem freien Steckplatz des Gehäuses. Detaillierte Informationen zur Installationsprozedur finden Sie im Servicehandbuch zu Ihrem System.
2. Schalten Sie die neue Platte ein. Geben Sie dazu in der Befehlszeile den folgenden Direktaufruf ein:
`smit ondisk`

Jetzt wird die Platte dem System hinzugefügt, aber sie ist noch nicht verfügbar. Die nächsten Schritte richten sich danach, ob die neue Platte Daten enthält.

- Wenn die Platte keine Daten enthält, fügen Sie sie mit einer der folgenden Methoden als physischen Datenträger zu einer Datenträgergruppe hinzu:
 - Geben Sie in der Befehlszeile den folgenden Direktaufruf ein, um die Platte zu einer vorhandenen Datenträgergruppe hinzuzufügen:
`smit extendvg`
 - Geben Sie in der Befehlszeile den folgenden Direktaufruf ein, um die Platte zu einer neuen Datenträgergruppe hinzuzufügen:
`smit mkvg`
- Wenn die Platte Daten enthält, importieren Sie die Daten.

Zugehörige Konzepte:

„Richtlinie für Datenträgergruppen implementieren“ auf Seite 442

Nachdem Sie entschieden haben, welche Richtlinien für die Datenträgergruppen Sie verwenden möchten, analysieren Sie Ihre aktuelle Konfiguration, indem Sie in der Befehlszeile den Befehl `lspv` eingeben.

Zugehörige Tasks:

„Eine Datenträgergruppe importieren oder exportieren“ auf Seite 383

Die folgende Tabelle erläutert, wie Sie durch Import und Export eine benutzerdefinierte Datenträgergruppe von einem System auf ein anderes verschieben. (Die Stammdatenträgergruppe `rootvg` kann nicht exportiert und importiert werden.)

„Eine Platte mit Daten entfernen“ auf Seite 423

Verwenden Sie diese Prozedur, um eine Platte mit Daten zu entfernen, ohne das System auszuschalten.

„Eine Platte ohne Daten entfernen“ auf Seite 424

Die folgende Prozedur beschreibt, wie Sie eine Platte entfernen, die keine Daten bzw. keine Daten enthält, die Sie aufbewahren möchten.

Namen eines logischen Datenträgers ändern:

Die folgende Prozedur beschreibt, wie Sie einen logischen Datenträger umbenennen, ohne Daten auf dem logischen Datenträger zu verlieren.

In den folgenden Beispielen wird der Name des logischen Datenträgers von `lv00` in `lv33` geändert.

1. Hängen Sie mit dem folgenden Befehl alle Dateisysteme ab, die dem logischen Datenträger zugeordnet sind:

```
umount /Name_des_Dateisystems
```

Name_des_Dateisystems steht für den vollständigen Namen eines Dateisystems.

Anmerkung:

- a. Der Befehl **umount** scheitert, wenn das Dateisystem, das Sie versuchen abzuhängen, derzeit im Gebrauch ist. Der Befehl **umount** wird nur ausgeführt, wenn keine Dateien im Dateisystem geöffnet sind und keine aktuellen Verzeichnisse von Benutzern auf dieser Einheit vorhanden sind.
 - b. Ein anderer Name für den Befehl **umount** ist **umount**. Die Namen sind gegeneinander austauschbar.
2. Benennen Sie den logischen Datenträger mit dem folgenden Befehl um:

```
chlv -n Neuer_LD-Name Alter_LD-Name
```

Das Flag **-n** gibt den neuen Namen für den logischen Datenträger (*Neuer_LD-Name*) an, und *Alter_LD-Name* steht für den Namen, den Sie ändern möchten. Beispiel:

```
chlv -n lv33 lv00
```

Anmerkung: Wenn Sie ein JFS- oder JFS2-Protokoll umbenennen, werden Sie vom System aufgefordert, den Befehl **chfs** für alle Dateisysteme auszuführen, die die umbenannte Protokolleinheit verwenden.

3. Hängen Sie die Dateisysteme, die Sie im Schritt 1 abgehängt haben, mit dem folgenden Befehl erneut an:

```
mount /test1
```

Jetzt ist der logische Datenträger umbenannt und verfügbar.

Einen logischen Datenträger auf einen anderen physischen Datenträger kopieren:

Je nach Anforderungen gibt es mehrere Möglichkeiten, um einen logischen Datenträger auf einen anderen physischen Datenträger zu kopieren und gleichzeitig die Integrität des Dateisystems zu bewahren.

Es gibt mehrere Methoden für das Kopieren eines logischen Datenträgers oder JFS auf einen anderen physischen Datenträger. Wählen Sie die Methode aus, die sich für Ihren Zweck am besten eignet.

Einen logischen Datenträger kopieren:

Wenn Sie den ursprünglichen logischen Datenträger kopieren und einen neuen logischen Datenträger auf dem physischen Zieldatenträger erstellen möchten, ist die einfachste Methode die Verwendung des Befehls **cplv**.

1. Verwenden Sie den logischen Datenträger nicht mehr. Hängen Sie das Dateisystem ab, und stoppen Sie alle Anwendungen, die auf den logischen Datenträger zugreifen.
2. Wählen Sie einen physischen Datenträger aus, der genügend Kapazität hat, um alle Daten aus dem ursprünglichen logischen Datenträger aufzunehmen.

Achtung: Wenn Sie Daten von einem größeren logischen Datenträger auf einen kleineren kopieren, können Sie damit das Dateisystem beschädigen, weil einige Daten (einschließlich des Superblocks) möglicherweise verloren gehen.

3. Kopieren Sie mit dem folgenden Befehl den ursprünglichen logischen Datenträger (in diesem Beispiel **lv00**), und erstellen Sie einen neuen:

Anmerkung: Der folgende Befehl **cplv** scheitert, wenn er einen neuen logischen Datenträger erstellt und die Datenträgergruppe im Parallelverarbeitungsmodus aktiviert ist.

```
cplv lv00
```

4. Hängen Sie die Dateisysteme an, und starten Sie die Anwendungen erneut, um mit der Verwendung des logischen Datenträgers zu beginnen.

Jetzt ist die Kopie des logischen Datenträgers verwendbar.

Einen logischen Datenträger bei weiterer Verfügbarkeit des ursprünglichen logischen Datenträgers kopieren:

Wenn Ihre Umgebung die ununterbrochene Verfügbarkeit des ursprünglichen Datenträgers erfordert, können Sie den Inhalt, wie im folgenden Beispiel gezeigt, mit dem Befehl **splitlvcopy** kopieren.

1. Verwenden Sie den folgenden SMIT-Direktaufruf, um den logischen Datenträger zu spiegeln:

```
smit mklvcopy
```

2. Verwenden Sie den logischen Datenträger nicht mehr. Hängen Sie das Dateisystem ab. Stoppen Sie alle Anwendungen, die auf den logischen Datenträger zugreifen, oder versetzen Sie sie in den Wartemodus.

Achtung: Im nächsten Schritt wird der Befehl **splitlvcopy** verwendet. Schließen Sie logische Datenträger, bevor Sie sie aufteilen, und hängen Sie alle enthaltenen Dateisysteme ab, bevor Sie diesen Befehl verwenden. Die Aufteilung eines geöffneten logischen Datenträgers kann Ihre Dateisysteme beschädigen und zum Verlust der Konsistenz zwischen dem ursprünglichen Datenträger und der Kopie führen, wenn mehrere Prozesse gleichzeitig auf den logischen Datenträger zugreifen.

3. Melden Sie sich als Root an, und kopieren Sie den ursprünglichen logischen Datenträger (**alterLD**) mit dem folgenden Befehl auf den neuen logischen Datenträger (**neuerLD**):

```
splitlvcopy -y neuerLD alterLD
```

Das Flag **-y** gibt den Namen des neuen logischen Datenträgers an. Wenn der für **alterLD** angegebene Datenträger keinen LVCB hat, wird der Befehl **splitlvcopy** zwar erfolgreich ausgeführt, aber es wird eine Nachricht generiert, die darauf hinweist, dass der für **neuerLD** angegebene Datenträger ohne LVCB erstellt wurde.

4. Hängen Sie die Dateisysteme an, und starten Sie die Anwendungen erneut, um mit der Verwendung des logischen Datenträgers zu beginnen.

Jetzt ist die Kopie des logischen Datenträgers verwendbar.

Einen logischen Datenträger ohne Dateisystem auf einen anderen physischen Datenträger kopieren:

Führen Sie die folgenden Schritte aus, um einen logischen Datenträger ohne Dateisystem auf einen anderen physischen Datenträger zu kopieren.

1. Erstellen Sie mit dem folgenden Befehl eine gespiegelte Kopie des logischen Datenträgers auf einem neuen physischen Datenträger in der Datenträgergruppe:

```
mk1vcopy Name_des_LogDat 2 Name_des_neuen_PhysDat
```

2. Synchronisieren Sie die Partitionen in der neuen Spiegelkopie mit dem folgenden Befehl:

```
syncvg -l Name_des_LogDat
```

3. Entfernen Sie die Kopie des logischen Datenträgers mit dem folgenden Befehl vom physischen Datenträger:

```
rm1vcopy Name_des_LogDat 1 Name_des_alten_PhysDat
```

Jetzt ist die Kopie des logischen Datenträgers ohne Dateisystem verwendbar.

Ein Dateisystemprotokoll auf einer dedizierten Platte für eine benutzerdefinierte Datenträgergruppe erstellen:

Ein JFS- oder JFS2-Dateisystemprotokoll ist eine formatierte Liste mit Datensätzen zu Dateisystemtransaktionen. Das Protokoll gewährleistet die Integrität des Dateisystems (aber nicht unbedingt die Datenintegrität), wenn das System abstürzt, aber die Transaktionen noch nicht abgeschlossen sind.

Bei der Installation des Systems wird eine dedizierte Platte auf hd8 für die Datenträgergruppe rootvg erstellt. Die folgende Prozedur unterstützt Sie beim Erstellen eines JFS-Protokolls auf einer separaten Platte für andere Datenträgergruppen. Wenn Sie ein JFS2-Protokoll erstellen, sind die folgenden Änderungen an der Prozedur vorzunehmen:

- Der Typ der Protokolleinheit ist jfs2log.
- Mit dem Befehl **logform** muss die Option **-V jfs2** zur Angabe einer JFS2-Protokolleinheit verwendet werden.
- Mit den Befehlen **crfs** muss jfs2 anstelle von jfs angegeben werden.

Anmerkung: Es besteht keine Voraussetzung, dass sich ein JFS2-Protokoll auf einer anderen Platte als das Dateisystem befindet. Es wird nur vorausgesetzt, dass sich die Protokolleinheiten in derselben Datenträgergruppe wie das Dateisystem befinden. In dieser Prozedur muss sich das JFS2-Protokoll aus Gründen der Leistungsverbesserung auf einer separaten Platte befinden.

Durch das Erstellen eines Dateisystemprotokolls für benutzerdefinierte Datenträgergruppen kann die Leistung unter bestimmten Bedingungen verbessert werden, wenn Sie einen NFS-Server haben und möchten, dass die Transaktionen für diesen Server protokolliert werden, ohne mit anderen Prozessen in Konkurrenz treten zu müssen.

Sie können die folgende Prozedur verwenden, die eine Datenträgergruppe (fsvg1) mit zwei physischen Datenträgern (hdisk1 und hdisk2) erstellt. Das Dateisystem befindet sich auf hdisk2 (ein 256-MB-Dateisystem, das über /u/myfs angehängt ist), und das Protokoll auf hdisk1. Ein JFS-Protokoll hat eine Standardgröße von 4 MB. Sie können selten verwendete Programme, wie z. B. /b1v, auf demselben physischen Datenträger wie das Protokoll installieren, ohne die Leistung zu beeinträchtigen.

Führen Sie zum Erstellen eines JFS-Protokolls für eine benutzerdefinierte Datenträgergruppe mit der SMIT- und der Befehlszeilenschnittstelle die folgenden Schritte aus:

1. Fügen Sie mit dem folgenden SMIT-Direktaufruf die neue Datenträgergruppe (in diesem Beispiel fsvg1) hinzu:

```
smit mkvg
```

2. Fügen Sie mit dem folgenden SMIT-Direktaufruf einen neuen logischen Datenträger zu dieser Datenträgergruppe hinzu:

```
smit mklv
```

3. Geben Sie in die folgenden Felder der Anzeige **Logischen Datenträger hinzufügen** Ihre Daten ein. Beispiel:

```

Name des logischen Datenträgers      fsvg1log
Anzahl der logischen Partitionen     1
Namen der physischen Datenträger     hdisk1
Typ des logischen Datenträgers       jfslog
Position auf dem physischen Datenträger center

```

4. Nachdem Sie die Felder ausgefüllt haben, drücken Sie die Eingabetaste, um Ihre Änderungen zu übernehmen. Beenden Sie SMIT.

5. Geben Sie den folgenden Befehl in einer Befehlszeile ein:

```
/usr/sbin/logform /dev/fsvg1log
```

6. Wenn die folgende Eingabeaufforderung erscheint, geben Sie y ein, und drücken Sie anschließend die Eingabetaste.

```
Destroy /dev/fsvg1log
```

Entgegen dem Wortlaut dieser Eingabeaufforderung wird nichts gelöscht. Wenn Sie mit y auf diese Eingabeaufforderung antworten, formatiert das System den logischen Datenträger für das JFS-Protokoll, damit Dateisystemtransaktionen aufgezeichnet werden können.

7. Fügen Sie mit dem folgenden SMIT-Direktaufruf einen weiteren logischen Datenträger hinzu:

```
smit mklv
```

8. Geben Sie den Namen derselben Datenträgergruppe ein, die Sie auch in Schritt 2 verwendet haben (in diesem Beispiel fsvg1). Geben Sie in die folgenden Felder der Anzeige "Logische Datenträger" Ihre Daten ein. Sie müssen für diesen logischen Datenträger einen anderen physischen Datenträger als in Schritt 3 angeben. Beispiel:

```

Name des logischen Datenträgers      fslv1
Anzahl der logischen Partitionen     64
Namen der physischen Datenträger     hdisk2
Typ des logischen Datenträgers       jfs

```

Nachdem Sie die Felder ausgefüllt haben, drücken Sie die Eingabetaste, um Ihre Änderungen zu übernehmen. Beenden Sie SMIT.

9. Verwenden Sie die folgende Befehlsfolge, um dem neuen logischen Datenträger ein Dateisystem hinzuzufügen, das Protokoll festzulegen und das neue Dateisystem anzuhängen:

```
crfs -v jfs -d Name_des_LD -m Dateisystemname -a logname=Pfad_des_DS-Protokolls
```

```
mount Dateisystemname
```

Name_des_LD steht für den Namen des logischen Datenträgers, den Sie in Schritt 2 erstellt haben, *Dateisystemname* für den Namen des Dateisystems, das Sie auf diesem logischen Datenträger erstellen möchten, und *Pfad_des_DS-Protokolls* für den Namen des logischen Datenträgers, den Sie in Schritt 2 erstellt haben. Beispiel:

```
crfs -v jfs -d fslv1 -m /u/myfs -a logname=/dev/fsvg1log
mount /u/myfs
```

10. Geben Sie den folgenden Befehl ein, um sicherzustellen, dass das Dateisystem und das Protokoll korrekt konfiguriert wurden. Sie müssen lediglich den Namen Ihrer Datenträgergruppe einsetzen:

```
lsvg -l fsvg1
```

Die Ausgabe zeigt die beiden von Ihnen erstellten logischen Datenträger zusammen mit dem jeweiligen Dateisystemtyp. Beispiel:

```

LV NAME          TYPE    ...
/dev/fsvg1log    jfslog ...
fslv1            jfs     ...

```

Sie haben eine Datenträgergruppe erstellt, die mindestens zwei logische Datenträger auf separaten physischen Datenträgern enthält, und einer dieser logischen Datenträger enthält das Dateisystemprotokoll.

Anmerkung: Aus Redundanzgründen können Sie die Spiegelung auf der Ebene der logischen Datenträger für die JFS2-Protokolleinheit zulassen. Die Spiegelung ist jedoch kein allgemein übliches Verfahren und nicht erforderlich.

Eine Datenträgergruppe importieren oder exportieren:

Die folgende Tabelle erläutert, wie Sie durch Import und Export eine benutzerdefinierte Datenträgergruppe von einem System auf ein anderes verschieben. (Die Stammdatenträgergruppe `rootvg` kann nicht exportiert und importiert werden.)

Die Exportprozedur entfernt die Definition einer Datenträgergruppe von einem System. Die Importprozedur führt die Datenträgergruppe in das neue System ein.

Sie können die Importprozedur auch verwenden, um eine Datenträgergruppe erneut im System einzufügen, wenn diese dem System zuvor bereits zugeordnet war, aber dann exportiert worden ist. Außerdem können Sie mit den Import- und Exportprozeduren einen physischen Datenträger, der Daten enthält, zu einer Datenträgergruppe hinzufügen, indem Sie die hinzuzufügende Platte in eine eigene Datenträgergruppe stellen.

Achtung: Der Befehl `importvg` ändert den Namen eines importierten logischen Datenträgers, wenn bereits ein logischer Datenträger mit diesem Namen im neuen System vorhanden ist. Wenn der Befehl `importvg` einen logischen Datenträger umbenennen muss, protokolliert er eine Fehlermeldung in der Standardfehlerausgabe. Liegen keine Konflikte vor, erstellt der Befehl `importvg` außerdem die Mountpunkte für die Dateien und Einträge in der Datei `/etc/filesystems`.

| Tasks für das Importieren und Exportieren von Datenträgergruppen | | |
|--|---|-------------------|
| Task | SMIT-Direktaufruf | Befehl oder Datei |
| Eine Datenträgergruppe importieren | <code>smit importvg</code> | |
| Eine Datenträgergruppe exportieren | <ol style="list-style-type: none"> 1. Dateisysteme auf logischen Datenträgern in der Datenträgergruppe abhängen: <code>smit umntdsk</code> 2. Datenträgergruppe abhängen: <code>smit varyoffvg</code> 3. Datenträgergruppe exportieren: <code>smit exportvg</code> | |

Achtung: Eine Datenträgergruppe, die einen Datenträger für den Paging-Bereich enthält, kann nicht exportiert werden, wenn der Paging-Bereich aktiv ist. Bevor Sie eine Datenträgergruppe mit einem aktiven Paging-Bereich exportieren, müssen Sie sicherstellen, dass der Paging-Bereich nicht automatisch während der Systeminitialisierung aktiviert wird. Geben Sie dazu den folgenden Befehl ein:

```
chps -a n paging_space Name
```

Führen Sie anschließend einen Warmstart durch, damit der Paging-Bereich inaktiv ist.

Zugehörige Tasks:

„Platten hinzufügen, während das System verfügbar bleibt“ auf Seite 378

Die folgende Prozedur beschreibt, wie Sie eine Platte mit der Funktion für das Entfernen von Einheiten im laufenden Betrieb aktivieren und konfigurieren. Diese Funktion ermöglicht Ihnen, Platten hinzuzufügen, ohne das System auszuschalten.

„Eine Platte mit Daten entfernen“ auf Seite 423

Verwenden Sie diese Prozedur, um eine Platte mit Daten zu entfernen, ohne das System auszuschalten.

Inhalt eines physischen Datenträgers migrieren:

Verwenden Sie die folgenden Anweisungen, um die physischen Partitionen, die zu einem oder mehreren der angegebenen logischen Datenträger gehören, von einem physischen Datenträger auf einen oder mehrere andere physische Datenträger innerhalb einer Datenträgergruppe zu verschieben. Sie können diese Prozedur auch verwenden, um Daten von einer fehlerhaften Platte zu verschieben, bevor Sie sie austauschen oder reparieren. Diese Prozedur kann auf physischen Datenträgern in der Stammdatenträgergruppe oder in einer benutzerdefinierten Datenträgergruppe verwendet werden.

Achtung: Wenn der logische Bootdatenträger von einem physischen Datenträger migriert wird, muss der Bootsatz auf dem Quellendatenträger gelöscht werden, da es ansonsten zu einer Systemblockierung kommen kann. Wenn Sie den Befehl **bosboot** ausführen, müssen Sie auch den Befehl **chpv -c** ausführen, der in Schritt 4 der folgenden Prozedur beschrieben wird.

1. Wenn Sie die Daten auf eine neue Platte migrieren möchten, führen Sie die folgenden Schritte aus. Andernfalls fahren Sie mit Schritt 2 fort.

a. Prüfen Sie, ob die Platte vom System erkannt wird und verfügbar ist. Geben Sie dazu Folgendes ein:

```
lsdev -Cc disk
```

Es erscheint eine Ausgabe wie die folgende:

```
hdisk0 Available 10-60-00-8,0 16 Bit LVD SCSI Disk Drive
hdisk1 Available 10-60-00-9,0 16 Bit LVD SCSI Disk Drive
hdisk2 Available 10-60-00-11,0 16 Bit LVD SCSI Disk Drive
```

b. Wenn die Platte aufgelistet wird und den Status "Available" (verfügbar) hat, prüfen Sie mit dem folgenden Befehl, ob die Platte nicht zu einer anderen Datenträgergruppe gehört:

```
lspv
```

Es erscheint eine Ausgabe wie die folgende:

```
hdisk0      0004234583aa7879      rootvg      active
hdisk1      00042345e05603c1      none        active
hdisk2      00083772caa7896e      imagesvg    active
```

In diesem Beispiel kann `hdisk1` als Ziellplatte verwendet werden, weil das dritte Feld zeigt, dass sie nicht von einer Datenträgergruppe verwendet wird.

Wenn die neue Platte nicht aufgelistet oder nicht verfügbar ist, müssen Sie den Plattenspeicher bzw. den Speicher für logische Datenträger konfigurieren.

c. Fügen Sie die neue Platte mit dem folgenden Befehl zur Datenträgergruppe hinzu:

```
extendvg Name_der_Datenträgergruppe Plattenname
```

Name_der_Datenträgergruppe steht für den Namen Ihrer Datenträgergruppe und *Plattenname* für den Namen der neuen Platte. In dem im vorherigen Schritt gezeigten Beispiel müssten Sie *Plattenname* durch `hdisk1` ersetzen.

2. Der physische Quellendatenträger und der physische Zieldatenträger müssen sich in derselben Datenträgergruppe befinden. Geben Sie Folgendes ein, um festzustellen, ob beide physischen Datenträger in der Datenträgergruppe enthalten sind:

```
lsvg -p Name_der_Datenträgergruppe
```

Name_der_Datenträgergruppe steht für den Namen Ihrer Datenträgergruppe. Für eine Stammdatenträgergruppe erscheint eine Ausgabe wie die folgende:

```
rootvg:
PV_NAME      PV STATE      TOTAL PPs   FREE PPs   FREE DISTRIBUTION
hdisk0       active        542         85         00..00..00..26..59
hdisk1       active        542         306        00..00..00..00..06
```

Achten Sie auf die für FREE PPs angegebene Zahl.

3. Prüfen Sie, ob auf der Zielplatte ausreichend Speicherplatz für den Quelldatenträger ist, dessen Inhalt Sie verschieben möchten:

- a. Ermitteln Sie mit dem folgenden Befehl die Anzahl der physischen Partitionen auf der Quellenplatte:

```
lspv Name_der_Quellenplatte | grep "USED PPs"
```

Name_der_Quellenplatte steht für den Namen der Quellenplatte, z. B. `hdisk0`. Es erscheint eine Ausgabe wie die folgende:

```
USED PPs:      159 (636 megabytes)
```

In diesem Beispiel benötigen Sie 159 freie physische Partitionen (FREE PPs) auf der Zielplatte, um die Migration erfolgreich durchführen zu können.

- b. Vergleichen Sie die Anzahl der verwendeten physischen Partitionen (USED PPs) auf der Quellenplatte mit der Anzahl der freien physischen Partitionen (FREE PPs) auf den Zielplatten (Schritt 2). Wenn die Anzahl freier physischer Partitionen größer ist als die Anzahl verwendeter physischer Partitionen, ist ausreichend Speicherplatz für die Migration verfügbar.
4. Führen Sie diesen Schritt nur aus, wenn Sie Daten von einer Platte in der Datenträgergruppe `rootvg` migrieren. Wenn Sie Daten von einer Platte in einer benutzerdefinierten Datenträgergruppe migrieren möchten, fahren Sie mit Schritt 5 fort.

Prüfen Sie mit dem folgenden Befehl, ob sich der logische Bootdatenträger (**hd5**) auf der Quellenplatte befindet:

```
lspv -l Nummer_der_Quellenplatte | grep hd5
```

Wenn Sie keine Ausgabe erhalten, befindet sich der logische Bootdatenträger nicht auf der Quellenplatte. Fahren Sie mit Schritt 5 fort.

Wenn Sie eine Ausgabe wie

```
hd5          2  2  02..00..00..00..00  /b1v
```

erhalten, führen Sie den folgenden Befehl aus:

```
migratepv -l hd5 Name_der_Quellenplatte Name_der_Zielplatte
```

Es wird eine Nachricht ausgegeben, in der Sie vor der Ausführung des Befehls **bosboot** auf der Zielplatte gewarnt werden. Sie müssen auch einen Befehl **mkboot -c** ausführen, um den Bootsatz auf der Quellenplatte zu löschen. Geben Sie die folgende Befehlsfolge ein:

```
bosboot -a -d /dev/Name_der_Zielplatte
bootlist -m normal Name_der_Zielplatte
mkboot -c -d /dev/Name_der_Quellenplatte
```

5. Migrieren Sie Ihre Daten mit dem folgenden SMIT-Direktaufruf:
`smit migratepv`
6. Listen Sie die physischen Datenträger auf, und wählen Sie den zuvor geprüften physischen Quelldatenträger aus.
7. Navigieren Sie zum Feld **Physischer Zieldatenträger**. Wenn Sie den Standardwert akzeptieren, sind alle physischen Datenträger in der Datenträgergruppe für die Übertragung verfügbar. Sie können aber auch eine oder mehrere Platten mit ausreichendem Speicherplatz für die zu verschiebenden Partitionen auswählen (Schritt 4).
8. Im Feld **Nur die zu diesem logischen Datenträger gehörenden Daten verschieben** können Sie logische Datenträger auflisten und einen Datenträger auswählen. In diesem Fall verschieben Sie nur die physischen Partitionen, die dem angegebenen logischen Datenträger zugeordnet sind und sich auf dem als physischer Quelldatenträger ausgewählten physischen Datenträger befinden.
9. Drücken Sie die Eingabetaste, um die physischen Partitionen zu verschieben.

Jetzt befinden sich die Daten auf der neuen (Ziel-)Platte. Die ursprüngliche (Quellen-)Platte bleibt jedoch in der Datenträgergruppe. Wenn die Platte noch zuverlässig ist, können Sie sie weiter als Hot-Spare-Platte verwenden. Insbesondere wenn eine Platte defekt ist, empfiehlt es sich, die folgenden Schritte auszuführen:

1. Geben Sie Folgendes ein, um die Quellenplatte aus der Datenträgergruppe zu entfernen:

```
reducevg Name_der_Datenträgergruppe Name_der_Quellenplatte
```

2. Geben Sie Folgendes ein, um die Quellenplatte physisch aus dem System zu entfernen:

```
rmdev -l Name_der_Quellenplatte -d
```

Zugehörige Konzepte:

„Speicherung logischer Datenträger“ auf Seite 409

Logische Datenträger sind Gruppen von Informationen auf physischen Datenträgern.

Zugehörige Tasks:

„Eine Platte konfigurieren“

Für die Konfiguration einer neuen Platte stehen mehrere Methoden zur Verfügung.

„Fehlerbehebung im Zusammenhang mit Plattenlaufwerken“ auf Seite 396

Im Folgenden wird beschrieben, wie Sie Probleme mit Plattenlaufwerken diagnostizieren und beheben.

Eine Platte konfigurieren:

Für die Konfiguration einer neuen Platte stehen mehrere Methoden zur Verfügung.

Sie können eine neue Platte mit einer der folgenden Methoden konfigurieren.

- Wenn Sie das System herunterfahren und ausschalten können, verwenden Sie Methode 1. Diese Methode ist, sofern anwendbar, die bevorzugte Methode für das Herunterfahren und Ausschalten jedes Systems, wenn Sie eine physische Platte anschließen.
- Wenn Sie das System nicht herunterfahren können und Details über die neue Platte kennen, z. B. die Unterklasse, den Typ, den übergeordneten Namen und die Anschlussposition, verwenden Sie Methode 2.
- Wenn Sie das System nicht herunterfahren können und nur die Position der Platte kennen, verwenden Sie Methode 3.

Nach der Konfiguration einer Platte erfordert der Logical Volume Manager, obwohl die Platte allgemein verfügbar ist, dass die Platte als physischer Datenträger identifiziert wird.

Methode 1

Verwenden Sie die folgende Methode, wenn Sie das System herunterfahren und ausschalten können, bevor Sie die Platte anschließen:

1. Schließen Sie die neue Platte physisch an das System an, und schalten Sie anschließend die Platte und das System gemäß den Anweisungen in der Dokumentation, die im Lieferumfang des Systems enthalten ist, ein.
2. Lassen Sie den Konfigurationsmanager (**cfgmgr**) beim Booten des Systems die Platte automatisch konfigurieren.
3. Geben Sie nach dem Booten des Systems in der Befehlszeile den Befehl **lspv** ein, um den Namen der neuen Platte zu suchen. Für die Ausführung dieses Befehls benötigen Sie Rootberechtigung. Das System gibt einen Eintrag ähnlich dem folgenden zurück:

```
hdisk1 none none
```

oder:

```
hdisk1 00005264d21adb2e none
```

Das erste Feld gibt den vom System zugeordneten Namen der Platte an. Das zweite Feld enthält die PVID (Physical Volume ID, ID des physischen Datenträgers), sofern vorhanden. Wenn die neue Platte nicht in der Ausgabe des Befehls **lspv** aufgeführt wird, ziehen Sie die Veröffentlichung *Installation and Migration* zu Rate.

Zu diesem Zeitpunkt ist die Platte für das System verwendbar, benötigt aber noch eine PVID für den LVM. Wenn die neue Platte keine PVID hat, lesen Sie die Informationen im Abschnitt „Eine verfügbare Platte zu einem physischen Datenträger machen“ auf Seite 388.

Methode 2

Verwenden Sie die folgende Methode, wenn Sie das System nicht herunterfahren können und die folgenden Informationen über die neue Platte kennen:

- Anschlusstyp der Platte (Unterklasse)
- Typ der Platte (Typ)
- Systemanschluss, an den die Platte angeschlossen ist (übergeordneter Name)
- Logische Adresse der Platte (des Anschlusses)

Gehen Sie wie folgt vor:

1. Schließen Sie die neue Platte physisch an das System an, und schalten Sie anschließend die Platte und das System gemäß den Anweisungen in der Dokumentation, die im Lieferumfang des Systems enthalten ist, ein.
2. Verwenden Sie den Befehl **mkdev**, wie im folgenden Beispiel gezeigt, mit den gezeigten Flags, um die Platte zu konfigurieren und sicherzustellen, dass sie als physischer Datenträger verfügbar ist:

```
mkdev -c disk -s scsi -t 2200mb -p scsi3 \  
-w 6,0 -a pv=yes
```

Dieser Beispielbefehl fügt eine 2,2-GB-Platte mit der SCSI-ID 6 und der Nummer 0 für die logische Einheit zum SCSI-Bus scsi3 hinzu. Das Flag **-c** definiert die Klasse der Einheit. Das Flag **-s** definiert die Unterklasse. Das Flag **-t** definiert den Typ der Einheit. Das Flag **-p** definiert den Namen der übergeordneten Einheit, den Sie zuordnen möchten. Das Flag **-w** bestimmt die Position der Platte mit SCSI-ID und Nummer der logischen Einheit. Das Flag **-a** gibt das Einheitenattribut/Wert-Paar (pv=yes) an, das aus der Platte einen physischen Datenträger macht, und schreibt einen Bootsatz mit einer eindeutigen PVID auf die Platte (sofern diese noch keine hat).

Jetzt ist die Platte als verfügbare Einheit und als physischer Datenträger definiert. Sie können den Befehl **lspv** in der Befehlszeile eingeben, um den neuen Platteneintrag aufzulisten. Wenn die neue Platte nicht in der Ausgabe des Befehls **lspv** aufgeführt wird, ziehen Sie die Veröffentlichung *Installation and Migration* zu Rate.

Methode 3

Verwenden Sie die folgende Methode, wenn Sie das System nicht herunterfahren können und nur die Position der Platte kennen:

1. Schließen Sie die neue Platte physisch an das System an und schalten Sie anschließend die Platte und das System gemäß den Anweisungen in der Dokumentation, die im Lieferumfang des Systems enthalten ist, ein.
2. Geben Sie in der Befehlszeile den Befehl **lspv** ein, um zu prüfen, welche physischen Platten bereits im System konfiguriert sind. Weitere Informationen zum Befehl **lspv** finden Sie im Abschnitt Befehl lspv. Es erscheint eine Ausgabe wie die folgende:

```
hdisk0      000005265ac63976  rootvg
```

3. Geben Sie in der Befehlszeile **cfgmgr** ein, um den Konfigurationsmanager aufzurufen. Der Konfigurationsmanager erkennt und konfiguriert automatisch alle neu angeschlossenen Einheiten im System,

einschließlich der neuen Platte. Weitere Informationen finden Sie in der Beschreibung des Befehls **cfgmgr** in der Veröffentlichung "Commands Reference".

4. Geben Sie den Befehl **lspv** erneut ein, um sicherzustellen, dass die neue Platte konfiguriert wurde. Es erscheint eine Ausgabe wie die folgende:

```
hdisk1    none                none
```

oder

```
hdisk1    00005264d21adb2e    none
```

Das erste Feld gibt den vom System zugeordneten Namen der Platte an. Das zweite Feld enthält die PVID (Physical Volume ID, ID des physischen Datenträgers), sofern vorhanden. Wenn die neue Platte nicht in der Ausgabe des Befehls **lspv** aufgeführt wird, ziehen Sie die Veröffentlichung *Installation and Migration* zu Rate.

Zu diesem Zeitpunkt ist die Platte für das System verwendbar, benötigt aber noch eine PVID für den LVM. Wenn die neue Platte keine PVID hat, lesen Sie die Informationen im Abschnitt „Eine verfügbare Platte zu einem physischen Datenträger machen“.

Zugehörige Tasks:

„Inhalt eines physischen Datenträgers migrieren“ auf Seite 384

Eine verfügbare Platte zu einem physischen Datenträger machen:

Eine Platte muss als physischer Datenträger konfiguriert werden, damit sie Datenträgergruppen zugeordnet und vom LVM verwendet werden kann.

Verwenden Sie die folgenden Anweisungen, um einen physischen Datenträger zu erstellen:

1. Vergewissern Sie sich, dass die Platte dem Betriebssystem bekannt ist, dass sie verfügbar ist und dass sie nicht vom Betriebssystem oder von anderen Anwendungen verwendet wird. Geben Sie in der Befehlszeile den Befehl **lspv** ein. Es erscheint eine Ausgabe wie die folgende:

```
hdisk1    none                none
```

Untersuchen Sie die Ausgabe wie folgt:

- Wenn der Name der neuen Platte nicht in der Befehlsausgabe erscheint, lesen Sie den Abschnitt „Eine Platte konfigurieren“ auf Seite 386.
- Wenn im zweiten Feld der Ausgabe eine vom System generierte PVID (Physical Volume Identifier, ID des physischen Datenträgers) angezeigt wird, z. B. 00005264d21adb2e, ist die Platte bereits als physischer Datenträger konfiguriert, und Sie müssen diese Prozedur nicht ausführen.
- Wenn im dritten Feld der Ausgabe der Name einer Datenträgergruppe angezeigt wird, z. B. rootvg, ist die Platte momentan im Gebrauch und kann für diese Prozedur nicht ausgewählt werden.

Wenn die neue Platte keine PVID hat und nicht im Gebrauch ist, fahren Sie mit dem nächsten Schritt fort.

2. Wenn Sie eine verfügbare Platte als physischen Datenträger konfigurieren möchten, geben Sie in der Befehlszeile den Befehl **chdev** ein: Beispiel:

```
chdev -l hdisk3 -a pv=yes
```

Das Flag **-l** gibt den Einheitennamen der Platte an. Das Flag **-a** gibt das Einheitenattribut/Wert-Paar (pv=yes), das aus der Platte einen physischen Datenträger macht, und schreibt einen Bootsatz mit einer eindeutigen PVID auf die Platte (sofern diese noch keine hat).

Jetzt ist die Platte als physischer Datenträger definiert. Sie können in der Befehlszeile den Befehl **lspv** eingeben, um den neuen Platteneintrag aufzulisten.

PVID und VGID von rootvg ändern:

Sie können die PVID (Physical Volume Identifier, ID des physischen Datenträgers) und die VGID (Volume Group Identifier, ID der Datenträgergruppe) der Datenträgergruppe "rootvg" (Stammdatenträgergruppe) während des Systemboots ändern.

Zum Ändern der PVID und VGID von rootvg definieren Sie das Attribut *sys0 dev ghostdev* mit dem Wert 2 und führen anschließend einen Warmstart des Systems durch. Das Attribut *sys0 device ghostdev* ist ein bitweises Flag.

- Geben Sie den folgenden Befehl ein, um das Attribut *sys0 device ghostdev* zum Ändern der PVID und VGID der Datenträgergruppe "rootvg" zu definieren:

```
chdev -l sys0 -a ghostdev=2
```

Anmerkung: Der Definition des Werts 2 für das Attribut *sys0 device ghostdev* wird aufgehoben, nachdem der Befehl **ipl_varyon** die PVID und VGID aller Platten in rootvg geändert hat. Wenn der Befehl **chdev** zum Ändern der PVID von rootvg-Platten fehlschlägt, sendet der Befehl **ipl_varyon** eine Warnung und fährt mit dem Anhängen von rootvg fort. Wenn der Befehl **chdev** zum Ändern der PVID von Platten in rootvg fehlschlägt und Sie die PVID und VGID beim nächsten Warmstart ändern möchten, setzen Sie das Attribut *sys0 device ghostdev* erneut auf den Wert 2.

- Zum Auflisten des Werts des Attributs *ghostdev* geben Sie den folgenden Befehl ein:

```
lsattr -E -l sys0 -a ghostdev
```

Einen fehlerhaften physischen Datenträger in einer gespiegelten Datenträgergruppe austauschen:

Die folgende Prozedur veranschaulicht, wie ein fehlerhafter physischer Datenträger in einer gespiegelten Datenträgergruppe ausgetauscht wird. Der Befehl **replacepv** ist eine Methode, mit der ein fehlerhafter physischer Datenträger in den meisten Konfigurationen ausgetauscht werden kann. Außerdem wird eine alternative Prozedur für Konfigurationen beschrieben, in denen der Befehl **replacepv** nicht verwendet werden kann.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

Voraussetzungen

- Alle logischen Datenträger, die den ausgefallenen physischen Datenträger verwenden, haben gültige Kopien auf anderen physischen Datenträgern (mit Ausnahme eines dedizierten logischen Speicherauszugsdatenträgers).

Einen ausgefallenen physischen Datenträger mit dem Befehl **replacepv** austauschen

Voraussetzungen

Falls eine der im Folgenden aufgelisteten Voraussetzungen nicht erfüllt werden kann, verwenden Sie die alternative Prozedur.

- Die Datenträgergruppe mit dem ausgefallenen physischen Datenträger ist nicht rootvg.
- Der physische Ersatzdatenträger kann der Datenträgergruppe mit dem ausgefallenen physischen Datenträger hinzugefügt werden (dies ist je nach Größe des physischen Datenträgers und Merkmalen der Datenträgergruppe, z. B. MAX PPs per PV unter Umständen nicht möglich).
- Der physische Ersatzdatenträger muss parallel zum defekten physischen Datenträger im System konfiguriert werden können.
- Der Name des physischen Ersatzdatenträgers kann anders sein als der des ausgefallenen physischen Datenträgers.
- Die Größe des physischen Ersatzdatenträgers muss kleiner-gleich der Größe des ausgefallenen physischen Datenträgers sein.

- Die Datenträgergruppe mit dem ausgefallenen physischen Datenträger darf keine Momentaufnahme sein und auch keine Momentaufnahme einer Datenträgergruppe enthalten.

Führen Sie die folgenden Schritte aus, in denen `hdisk2` der ausgefallene physische Datenträger und `hdisk10` der physische Ersatzdatenträger ist:

1. Wenn der physische Ersatzdatenträger noch nicht im System installiert ist, führen Sie die erforderlichen Installationsschritte aus. Führen Sie den folgenden Befehl aus, wenn Sie mit dem Konfigurationsmanager einen neuen physischen Datenträger definieren möchten:

```
cfgmgr
```

Verwenden Sie den Befehl `lspv`, um den Namen festzustellen, der dem physischen Datenträger zugeordnet ist. Für dieses Beispiel wird angenommen, dass der neue physische Datenträger den Namen `hdisk10` hat.

2. Führen Sie den folgenden Befehl aus, um den ausgefallenen physischen Datenträger durch den Datenträger zu ersetzen, den Sie in Schritt 1 definiert haben:

```
replacepv hdisk2 hdisk10
```

Wenn Sie diesen Befehl ausführen, wird `hdisk2` durch `hdisk10` ersetzt, und `hdisk2` ist der Datenträgergruppe nicht mehr zugeordnet.

3. Führen Sie den folgenden Befehl aus, um die Definition des ausgefallenen physischen Datenträgers aufzuheben:

```
rmdev -d1 hdisk2
```

4. Entfernen Sie die ausgefallene Platte physisch aus dem System.

5. Führen Sie die folgenden Schritte aus, um sich zu vergewissern, dass die Prozedur erfolgreich war:

- Führen Sie den folgenden Befehl aus, um zu prüfen, ob alle logischen Datenträger wie gewünscht auf den neuen physischen Datenträger gespiegelt wurden:

```
lslv Name_des_logischen_Datenträgers
```

Überprüfen Sie das Attribut `COPIES` jedes logischen Datenträgers, der vom ausgefallenen physischen Datenträger betroffen ist, um sicherzustellen, dass die gewünschte Anzahl von Kopien vorhanden ist. Wenn die Anzahl der Kopien des logischen Datenträgers unter der gewünschten Anzahl liegt, erstellen Sie mit dem Befehl `mkivcopy` zusätzliche Kopien.

- Führen Sie den folgenden Befehl aus, um sicherzustellen, dass alle logischen Datenträgerpartitionen synchronisiert sind und keine veralteten Partitionen vorhanden sind:

```
lspv hdisk10
```

Vergewissern Sie sich, dass das Attribut `STALE PARTITIONS` des ausgetauschten physischen Datenträgers den Wert `null` hat. Sollten veraltete Partitionen vorhanden sein, synchronisieren Sie die Partitionen mit dem Befehl `syncvg`.

Mit Schritt 5 ist das Austauschen eines ausgefallenen physischen Datenträgers abgeschlossen.

Einen ausgefallenen physischen Datenträger austauschen, wenn der Befehl `replacepv` laut Konfiguration nicht zulässig ist

Angenommen, der ausgefallene physische Datenträger `hdisk0` und seine Spiegelung `hdisk1` gehören zur Datenträgergruppe `yourvg`.

1. Führen Sie den folgenden Befehl aus, um Spiegelkopien vom ausgefallenen physischen Datenträger zu entfernen:

```
unmirrorvg yourvg hdisk0
```

2. Wenn der ausgefallene physische Datenträger zur Stammdatenträgergruppe `rootvg` gehört, entfernen Sie `hdisk0` mit dem folgenden Befehl aus der Bootliste:

Anmerkung: Wenn in Ihrer Konfiguration andere Booteinheiten als `hdisk0` und `hdisk1` verwendet werden, fügen Sie diese zur Befehlssyntax hinzu.

```
bootlist -om normal hdisk1
```

Dieser Schritt setzt voraus, dass `hdisk1` eine bootfähige Einheit in `rootvg` bleibt. Stellen Sie nach der Ausführung dieses Schritts sicher, dass `hdisk0` nicht mehr in der Ausgabe erscheint.

3. Wenn der ausgefallene physische Datenträger zur Stammdatenträgergruppe `rootvg` gehört, erstellen Sie alle dedizierten Speicherauszugseinheiten, die auf dem ausgefallenen physischen Datenträger enthalten sind, erneut.

Wenn eine dedizierte Speicherauszugseinheit auf dem ausgefallenen physischen Datenträger vorhanden ist, können Sie mit dem Befehl **`mklv`** einen neuen logischen Datenträger auf einem vorhandenen physischen Datenträger erstellen. Verwenden Sie den Befehl **`sysdumpdev`**, um den neuen logischen Datenträger als primäre Speicherauszugseinheit festzulegen.

4. Führen Sie den folgenden Befehl aus, um die Definition des ausgefallenen physischen Datenträgers aufzuheben:

Anmerkung: Wenn Sie den Eintrag für die Platteneinheit entfernen, wird auch die feste Verbindung `/dev/ipldevice` entfernt, falls der ausgefallene physische Datenträger der für das Booten des Systems verwendete physische Datenträger ist.

```
reducevg yourvg hdisk0  
rmdev -dl hdisk0
```

5. Wenn der ausgefallene physische Datenträger die zuletzt verwendete Booteinheit ist, erstellen Sie die feste Verbindung `/dev/ipldevice`, die in Schritt 4 entfernt wurde, mit dem folgenden Befehl erneut:

```
ln /dev/rhdisk1 /dev/ipldevice
```

Beachten Sie das `r` vor dem Namen des physischen Datenträgers.

Führen Sie den folgenden Befehl aus, um sicherzustellen, dass die feste Verbindung `/dev/ipldevice` erneut erstellt worden ist:

```
ls /dev/ipldevice
```

6. Ersetzen Sie die ausgefallene Platte.
7. Führen Sie den folgenden Befehl aus, um den neuen physischen Datenträger zu definieren:

```
cfgmgr
```

Der Befehl **`cfgmgr`** ordnet dem physischen Ersatzdatenträger einen Namen zu. Wahrscheinlich ist der Name, der dem neuen physischen Datenträger zugeordnet wird, identisch mit dem Namen, der zuvor dem ausgefallenen physischen Datenträger zugeordnet war. In diesem Beispiel wird angenommen, dass dem physischen Ersatzdatenträger die Einheit `hdisk0` zugeordnet wird.

8. Führen Sie den folgenden Befehl aus, um den neuen physischen Datenträger zur Datenträgergruppe hinzuzufügen:

```
extendvg yourvg hdisk0
```

Die folgende Fehlermeldung kann ausgegeben werden:

```
0516-050 Es ist nicht genügend Platz im Deskriptorbereich der Datenträgergruppe vorhanden.  
Fügen Sie einen kleineren physischen Datenträger hinzu oder verwenden Sie eine andere Datenträgergruppe.
```

Wenn diese Fehlermeldung ausgegeben wird und der physische Datenträger nicht zur Datenträgergruppe hinzugefügt werden kann, können Sie versuchen, logische Datenträger auf einem anderen physischen Datenträger zu spiegeln, der bereits in der Datenträgergruppe vorhanden ist, oder einen kleineren physischen Datenträger hinzufügen. Wenn beide Optionen nicht möglich sind, können Sie versuchen, diese Einschränkung zu umgehen, indem Sie die Datenträgergruppe mit dem Befehl **`chvg`** auf eine große oder skalierbare Datenträgergruppe aufrüsten.

9. Spiegeln Sie die Datenträgergruppe.

Anmerkung: Der Befehl **mirrorvg** kann nicht verwendet werden, wenn alle der folgenden Bedingungen zutreffen:

- Das Zielsystem ist eine logische Partition (LPAR).
- Auf dem ausgefallenen physischen Datenträger befindet sich eine Kopie des logischen Bootdatenträgers (standardmäßig hd5).
- Der Adapter des physischen Ersatzdatenträgers wurde nach dem letzten Kaltstart dynamisch in der LPAR konfiguriert.

Wenn alle genannten Bedingungen zutreffen, erstellen Sie mit dem Befehl **mklvcopy** wie folgt Spiegelkopien für jeden logischen Datenträger:

- a. Erstellen Sie Kopien des logischen Bootdatenträgers, um sicherzustellen, dass er einer zusammenhängenden Gruppe physischer Partitionen zugeordnet wird.
- b. Erstellen Sie Kopien der verbleibenden logischen Datenträger, und synchronisieren Sie die Kopien mit dem Befehl **syncvg**.
- c. Machen Sie die Platte bootfähig, indem Sie die LPAR herunterfahren und wieder aktivieren, anstatt mit den Befehlen shutdown oder reboot einen Warmstart durchzuführen. Die LPAR muss nicht sofort heruntergefahren werden, aber das System muss auf jeden Fall von dem neuen physischen Datenträger gebootet werden.

Erstellen Sie andernfalls neue Kopien der logischen Datenträger in der Datenträgergruppe. Geben Sie dazu den neuen physischen Datenträger mit dem folgenden Befehl an:

Anmerkung: Der Befehl **mirrorvg** inaktiviert das Quorum standardmäßig. Für die Stammdatenträgergruppe (rootvg) können Sie die Option **-m** verwenden, um sicherzustellen, dass die neuen logischen Datenträgerkopien `hdisk0` auf dieselbe Weise zugeordnet werden wie die funktionierende Platte.

```
mirrorvg yourvg hdisk0
```

10. Wenn Ihre Konfiguration Kopien von logischen Datenträgern enthält, müssen Sie diese Kopien unter Umständen mit dem folgenden Befehl erneut erstellen:

```
mklvcopy -k
```

11. Wenn der ausgefallene physische Datenträger zur Stammdatenträgergruppe rootvg gehört hat, initialisieren Sie den Bootsatz mit dem folgenden Befehl:

```
bosboot -a
```

12. Wenn der ausgefallene physische Datenträger zur Stammdatenträgergruppe rootvg gehört hat, aktualisieren Sie die Bootliste mit dem folgenden Befehl:

Anmerkung: Wenn in Ihrer Konfiguration andere Booteinheiten als `hdisk0` und `hdisk1` verwendet werden, fügen Sie diese dem Befehl hinzu.

```
bootlist -om normal hdisk0 hdisk1
```

13. Vergewissern Sie sich, dass die Prozedur erfolgreich war.

- Führen Sie den folgenden Befehl aus, um zu prüfen, ob alle logischen Datenträger auf dem neuen physischen Datenträger gespiegelt wurden:

```
lslv Name_des_Logischen_Datenträgers
```

Überprüfen Sie das Attribut COPIES jedes logischen Datenträgers, der vom ausgefallenen physischen Datenträger betroffen ist, um sicherzustellen, dass die gewünschte Anzahl von Kopien vorhanden ist. Wenn die Anzahl der Kopien des logischen Datenträgers unter der gewünschten Anzahl liegt, erstellen Sie mit dem Befehl **mklvcopy** zusätzliche Kopien.

- Führen Sie den folgenden Befehl aus, um sicherzustellen, dass alle logischen Datenträgerpartitionen synchronisiert sind und keine veralteten Partitionen vorhanden sind:

```
lspv hdisk0
```


Vergewissern Sie sich, dass das Attribut `STALE PARTITIONS` des ausgetauschten physischen Datenträgers den Wert `null` hat. Sollten veraltete Partitionen vorhanden sein, synchronisieren Sie die Partitionen mit dem Befehl `syncvg`.

Wenn der ausgefallene physische Datenträger zur Stammdatenträgergruppe `rootvg` gehört hat, führen Sie die folgenden Schritte aus, um weitere Aspekte dieser Prozedur zu prüfen:

- Führen Sie den folgenden Befehl aus, um die Bootliste zu prüfen:
`bootlist -om normal`
- Führen Sie den folgenden Befehl aus, um die Speicherauszugseinheit zu prüfen:
`sysdumpdev -l`
- Führen Sie den folgenden Befehl aus, um die Liste der bootfähigen physischen Datenträger zu prüfen:
`ipl_varyon -i`
- Führen Sie den folgenden Befehl aus, um `/dev/ipl_device` zu prüfen:
`ls -i /dev/rhdisk1 /dev/ipldevice`

Vergewissern Sie sich, dass die Ausgabe des Befehls `ls` für beide Einträge dieselbe I-Node-Nummer enthält.

Mit diesem Schritt ist die Prozedur abgeschlossen.

Zugehörige Informationen:

 [Logical Volume Manager from A to Z: Introduction and Concepts](#)

Administrator beim Fehlen eines physischen Datenträgers benachrichtigen:

Obwohl AIX einen Fehler protokolliert, wenn ein physischer Datenträger nicht mehr zugänglich ist, kann ein Fehler in bestimmten Situationen unerkannt bleiben.

Wenn der physische Datenträger beispielsweise zu einer gespiegelten Datenträgergruppe gehört, bemerken die Benutzer dieses Problem nicht, weil weiterhin eine fehlerfreie Kopie der Daten zugänglich ist. In solchen Fällen kann der Administrator durch automatische Benachrichtigung auf das Problem hingewiesen werden, bevor die Benutzer in ihrer Arbeit gestört werden.

Die folgende Prozedur beschreibt, wie Sie die automatische Benachrichtigung für den Fall konfigurieren, dass ein physischer Datenträger als fehlend deklariert wird. Durch Abändern der folgenden Prozedur können Sie andere, Ihnen wichtige Fehler überwachen.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Mit Rootberechtigung können Sie eine Sicherungskopie der ODM-Datei `/etc/objrepos/errnotify` erstellen. Sie können die Sicherungskopie beliebig benennen. Im folgenden Beispiel hat die Sicherungskopie den Namen der Datei `errnotify`, dem das aktuelle Datum angefügt wird:

```
cd /etc/objrepos
cp errnotify errnotifyaktuelles_Datum
```

2. Erstellen Sie mit Ihrem Editor eine Datei mit dem Namen `/tmp/pvmiss.add`, die die folgende Zeilen-Gruppe enthält:

```
errnotify:
en_pid = 0
en_name = "LVM_SA_PVMISS"
en_persistenceflg = 1
en_label = "LVM_SA_PVMISS"
en_crcid = 0
en_type = "UNKN"
en_alertflg = ""
en_resource = "LVDD"
```

```

en_rtype = "NONE"
en_rclass = "NONE"
en_method = "/usr/lib/ras/pvmiss.notify $1 $2 $3 $4 $5 $6 $7 $8 $9"

```

Nachdem Sie alle in diesem Abschnitt beschriebenen Schritte ausgeführt haben, erweitert der Dämon für Fehlerbenachrichtigung automatisch \$1 bis \$9 in diesem Script mit detaillierten Informationen aus dem Fehlerprotokolleintrag in der Hinweismeldung.

- Erstellen Sie mit Ihrem Editor eine Datei mit dem Namen `/usr/lib/ras/pvmiss.notify` mit dem folgenden Inhalt:

```

#!/bin/ksh
exec 3>/dev/console
print -u3 "?"
print -u3 - "-----"
print -u3 "ALERT! ALERT! ALERT! ALERT! ALERT! ALERT!"
print -u3 ""
print -u3 "Desc: PHYSICAL VOLUME IS MISSING. SEE ERRPT."
print -u3 ""
print -u3 "Error label: $9"
print -u3 "Sequence number: $1"
print -u3 "Error ID: $2"
print -u3 "Error class: $3"
print -u3 "Error type: $4"
print -u3 "Resource name: $6"
print -u3 "Resource type: $7"
print -u3 "Resource class: $8"
print -u3 - "-----"
print -u3 "?"
mail - "PHSYICAL VOLUME DECLARED MISSING" root <<-EOF
-----
ALERT! ALERT! ALERT! ALERT! ALERT! ALERT!
Desc: PHYSICAL VOLUME IS MISSING. SEE ERRPT.
Error label: $9
Sequence number: $1
Error ID: $2
Error class: $3
Error type: $4
Resource name: $6
Resource type: $7
Resource class: $8
-----
EOF

```

- Speichern Sie Ihre Datei und beenden Sie den Editor.
- Legen Sie die Berechtigungen für die soeben erstellte Datei fest. Beispiel:

```
chmod 755 /usr/lib/ras/pvmiss.notify
```
- Geben Sie den folgenden Befehl ein, um die `LVM_SA_PVMISS`-Definition, die Sie in Schritt 2 erstellt haben, zu ODM hinzuzufügen:

```
odmadd /tmp/pvmiss.add
```

Jetzt führt das System das Script `/usr/lib/ras/pvmiss.notify` jedes Mal aus, wenn ein Fehler vom Typ `LVM_SA_PVMISS` auftritt. Dieses Script sendet eine Nachricht an die Konsole und außerdem eine E-Mail an Root.

Zugehörige Konzepte:

„Speicherung logischer Datenträger“ auf Seite 409

Logische Datenträger sind Gruppen von Informationen auf physischen Datenträgern.

Zugehörige Informationen:

odmadd command

Einen gespiegelten Datenträger von einer Datenträgergruppe abtrennen:

Sie können durch *Momentaufnahmen* (oder Snapshots) die Konsistenz gespiegelter Datenträgergruppen schützen und damit potenziellen Plattenausfällen vorbeugen.

Mit der Funktion für Momentaufnahme können Sie gespiegelte Datenträger abtrennen, um sie als zuverlässige (im Bezug auf die LVM-Metadaten) Sicherungen einer Datenträgergruppe mit Zeitangabe zu verwenden, und bei Bedarf die abgetrennten Datenträger wieder in die Datenträgergruppe integrieren. In der folgenden Prozedur trennen Sie zuerst einen gespiegelten Datenträger von einer Datenträgergruppe ab und integrieren danach den abgetrennten Datenträger wieder in die ursprüngliche Datenträgergruppe. Um die Zuverlässigkeit der Momentaufnahme zu gewährleisten, müssen die Dateisysteme abgehängt werden. Außerdem müssen Anwendungen, die logische Datenträger ohne Dateisystem verwenden, einen bekannten Status haben (einen Status, in dem die Anwendung wiederhergestellt werden kann, falls Sie die Sicherung verwenden müssen).

Eine Datenträgergruppe kann nicht aufgeteilt werden, falls eine der folgenden Bedingungen zutrifft:

- Es fehlt bereits ein Datenträger.
- Die letzte nicht veraltete Partition würde sich in der aufgeteilten Datenträgergruppe befinden.
- Es sind veraltete Partitionen in der Datenträgergruppe vorhanden (gilt nicht, wenn Sie das Flag **-f** (für "force", erzwingen) mit dem Befehl **splitvg** angeben).

Die Funktion für Momentaufnahme (insbesondere der Befehl **splitvg**) kann außerdem im erweiterten und im klassischen Parallelverarbeitungsmodus nicht verwendet werden. Die aufgeteilte Datenträgergruppe kann nicht für den Parallel- oder erweiterten Parallelverarbeitungsmodus aktiviert werden. Ferner sind Einschränkungen bezüglich der Änderungen, die an der aufgeteilten und an der ursprünglichen Datenträgergruppe vorgenommen werden können, zu beachten. Einzelheiten hierzu finden Sie in der Beschreibung des Befehls **chvg**.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Vergewissern Sie sich, dass die Datenträgergruppe vollständig gespiegelt ist und dass die Spiegelung auf Datenträger geschrieben wurde, die nichts anderes enthalten.
2. Zur Aktivierung der Unterstützung für Momentaufnahme teilen Sie die ursprüngliche Datenträgergruppe (**origVG**) mit dem folgenden Befehl auf andere Datenträger auf:

```
splitvg origVG
```

Damit haben Sie eine zuverlässige Sicherung der ursprünglichen Datenträgergruppe mit Zeitangabe. Die Zuordnung der aufgeteilten Datenträgergruppe kann jedoch nicht geändert werden.

3. Reaktivieren Sie den abgetrennten Datenträger und integrieren Sie ihn mit dem folgenden Befehl wieder in die ursprüngliche Datenträgergruppe:

```
joinvg origVG
```

Damit ist die aufgeteilte Datenträgergruppe wieder in die ursprüngliche Datenträgergruppe integriert.

Zugehörige Konzepte:

„Speicherung logischer Datenträger“ auf Seite 409

Logische Datenträger sind Gruppen von Informationen auf physischen Datenträgern.

Zugehörige Informationen:

`chvg` command

`recreatevg` command

`splitvg` command

➡ Logical Volume Manager from A to Z: Introduction and Concepts

Fehlerbehebung im Zusammenhang mit dem LVM

Es gibt verschiedene allgemeine Typen von Problemen mit dem LVM, die Sie beheben können.

Fehlerbehebung im Zusammenhang mit Plattenlaufwerken:

Im Folgenden wird beschrieben, wie Sie Probleme mit Plattenlaufwerken diagnostizieren und beheben.

Wenn Sie vermuten, dass ein Plattenlaufwerk einen mechanischen Defekt hat oder ausgefallen ist, führen Sie wie folgt die Plattendiagnose durch:

1. Melden Sie sich als Benutzer mit Rootberechtigung an, und geben Sie in der Befehlszeile den folgenden SMIT-Direktaufruf ein:
smit diag
2. Wählen Sie **Diagnoseprogramme in der aktuellen Shell** aus, um das Diagnosetool von AIX aufzurufen.
3. Drücken Sie die Eingabetaste, nachdem Sie die Anzeige "Diagnose-Operations-Anweisungen" gelesen haben.
4. Wählen Sie **Diagnoseroutinen** aus.
5. Wählen Sie **Systemprüfung** aus.
6. Blättern Sie die Liste durch, bis Sie das Laufwerk finden, das Sie testen möchten. Wählen Sie dieses Laufwerk aus.
7. Wählen Sie **Festschreiben** aus.

Auf der Basis der Diagnoseergebnisse sollten Sie in der Lage sein, den Zustand der Platte zu bestimmen:

- Wenn Sie feststellen, dass das Plattenlaufwerk defekt oder ausgefallen ist, liegt das Hauptaugenmerk auf der Wiederherstellung der Daten auf dieser Platte. Migration ist die bevorzugte Methode für die Wiederherstellung von Daten von einer defekten Platte. Die folgenden Prozeduren beschreiben, wie Sie Daten auf logischen Datenträgern wiederherstellen oder zurückschreiben, wenn eine Migration nicht erfolgreich durchgeführt werden kann.
- Wenn Ihr Laufwerk defekt ist und Sie das Laufwerk reparieren können, ohne es zu formatieren, gehen keine Daten verloren.
- Wenn das Plattenlaufwerk erneut formatiert oder ausgetauscht werden muss, erstellen Sie, sofern möglich, eine Sicherung, und entfernen Sie das Plattenlaufwerk aus seiner Datenträgergruppe und aus der Systemkonfiguration, bevor Sie es austauschen. Einige Daten aus Dateisystemen mit nur einer Kopie können verloren gehen.

Zugehörige Konzepte:

„Speicherplatz auf Plattenlaufwerken“ auf Seite 397

Wenn der Speicherplatz auf einem Plattenlaufwerk knapp wird, können Sie dieses Problem auf verschiedene Arten beheben. Sie können nicht erwünschte Dateien automatisch überwachen und entfernen, den Zugriff auf bestimmte Verzeichnisse für Benutzer einschränken oder Speicherplatz von einem anderen Plattenlaufwerk anhängen.

„Wiederherstellung von Plattenlaufwerken ohne erneute Formatierung“ auf Seite 398

Wenn Sie eine defekte Platte reparieren und anschließend wieder im System installieren, ohne sie erneut zu formatieren, können Sie die veralteten physischen Partitionen auf dem Laufwerk während des Bootvorgangs automatisch vom System aktivieren und resynchronisieren lassen. Eine *veraltete physische Partition* enthält Daten, die Ihr System nicht verwenden kann.

Zugehörige Tasks:

„Inhalt eines physischen Datenträgers migrieren“ auf Seite 384

„Wiederherstellung mit einem erneut formatierten oder Ersatzplattenlaufwerk“ auf Seite 398

Sie können Daten von einem defekten Plattenlaufwerk wiederherstellen, wenn Sie die defekte Platte erneut formatieren oder austauschen müssen.

„Wiederherstellung nach einem Plattenausfall, während das System verfügbar bleibt“ auf Seite 404
Die Wiederherstellung nach einem Plattenausfall ist mit der Funktion für das Entfernen von Einheiten bei laufendem Betrieb möglich.

Speicherplatz auf Plattenlaufwerken:

Wenn der Speicherplatz auf einem Plattenlaufwerk knapp wird, können Sie dieses Problem auf verschiedene Arten beheben. Sie können nicht erwünschte Dateien automatisch überwachen und entfernen, den Zugriff auf bestimmte Verzeichnisse für Benutzer einschränken oder Speicherplatz von einem anderen Plattenlaufwerk anhängen.

Für die Ausführung dieser Tasks müssen Sie die Berechtigungen von Root, der Gruppe system oder einer Administrationsgruppe besitzen.

Zugehörige Tasks:

„Fehlerbehebung im Zusammenhang mit Plattenlaufwerken“ auf Seite 396

Im Folgenden wird beschrieben, wie Sie Probleme mit Plattenlaufwerken diagnostizieren und beheben.

Befehl für die automatische Bereinigung von Dateisystemen:

Mit dem Befehl **skulker** können Sie Dateisysteme bereinigen und die nicht erwünschten Dateien entfernen.

Geben Sie in der Befehlszeile Folgendes ein:

```
skulker -p
```

Der Befehl **skulker** wird verwendet, um in regelmäßigen Abständen veraltete oder nicht mehr benötigte Dateien aus Dateisystemen zu löschen. Zu den Kandidaten gehören Dateien im Verzeichnis /tmp, Dateien, die älter sind als das angegebene Alter, Dateien a.out, Kerndateien und Dateien vom Typ ed.hup. Weitere Informationen finden Sie in der Beschreibung des Befehls **skulker** in der Veröffentlichung "Commands Reference".

Der Befehl **skulker** wird in der Regel täglich im Rahmen einer Abrechnungsprozedur ausgeführt, die der Befehl **cron** in Zeiten geringer Systemauslastung ausgeführt.

Zugehörige Konzepte:

„Plattenüberlauf“ auf Seite 478

Ein Plattenüberlauf tritt auf, wenn zu viele Dateien den reservierten Speicherplatz füllen. Der Grund hierfür kann ein "Ausreißerprozess" sein, der viele nicht benötigte Dateien erstellt.

Zugehörige Tasks:

„Abrechnungssystem konfigurieren“ auf Seite 176

Sie können ein Abrechnungssystem konfigurieren.

Zugriff der Benutzer auf bestimmte Verzeichnisse einschränken:

Sie können Plattenspeicherplatz freigeben und diesen unter Umständen frei halten, indem Sie den Zugriff auf Verzeichnisse einschränken und die Plattenbelegung überwachen.

1. Geben Sie den folgenden Befehl ein, um den Zugriff der Benutzer auf bestimmte Verzeichnisse einzuschränken:

```
chmod 755 Verzeichnisname
```

Dieser Befehl setzt Lese- und Schreibberechtigungen für den Eigner (Root) und Leseberechtigungen für die Gruppe und andere. *Verzeichnisname* steht für den vollständigen Pfadnamen des Verzeichnisses, für das Sie den Zugriff einschränken möchten.

2. Überwachen Sie die Plattenbelegung einzelner Benutzer, indem Sie der Datei /var/spool/cron/crontabs/**adm** die folgende Zeile hinzufügen:

```
0 2 * * 4 /usr/sbin/acct/dodisk
```

Diese Zeile führt den Befehl **dodisk** jeden Donnerstag (4) um 2 Uhr morgens (0 2) aus. Der Befehl **dodisk** leitet die Plattenbelegungsabrechnung ein. Dieser Befehl wird gewöhnlich im Rahmen einer vom Befehl **cron** ausgeführten Abrechnungsprozedur zu Zeiten geringer Systemauslastung ausgeführt.

Zugehörige Tasks:

„Abrechnungssystem konfigurieren“ auf Seite 176
Sie können ein Abrechnungssystem konfigurieren.

Speicherplatz von einem anderen Plattenlaufwerk anhängen:

Sie können mehr Speicherplatz auf einem Plattenlaufwerk anfordern, indem Sie Speicherplatz von einem anderen Laufwerk anhängen.

Sie können die folgenden Methoden verwenden, um Speicherplatz von einer anderen Platte anzuhängen:

- Direktaufruf **smit mountfs**,
- Befehl **mount**. Beispiel:

```
mount -n KnotenA -vnfs /usr/spool /usr/meinspool
```

Der Befehl **mount** macht ein Dateisystem an einer bestimmten Position verfügbar.

Zugehörige Verweise:

„Dateisysteme verwalten“ auf Seite 466

Die einfachsten Tasks, die Sie für die Verwaltung von Dateisystemen benötigen, sind in der folgenden Tabelle gruppiert.

Wiederherstellung von Plattenlaufwerken ohne erneute Formatierung:

Wenn Sie eine defekte Platte reparieren und anschließend wieder im System installieren, ohne sie erneut zu formatieren, können Sie die veralteten physischen Partitionen auf dem Laufwerk während des Bootvorgangs automatisch vom System aktivieren und resynchronisieren lassen. Eine *veraltete physische Partition* enthält Daten, die Ihr System nicht verwenden kann.

Wenn Sie vermuten, dass eine physische Partition veraltet ist, geben Sie in der Befehlszeile Folgendes ein:

```
lspv -M PhysDatName
```

PhysDatName steht für den Namen des physischen Datenträgers. In der Ausgabe des Befehls **lspv** werden alle Partitionen auf dem physischen Datenträger aufgelistet. Im Folgenden sehen Sie einen Auszug aus einer Beispielausgabe:

| | | |
|-------------|-----------|-------|
| hdisk16:112 | lv01:4:2 | stale |
| hdisk16:113 | lv01:5:2 | stale |
| hdisk16:114 | lv01:6:2 | stale |
| hdisk16:115 | lv01:7:2 | stale |
| hdisk16:116 | lv01:8:2 | stale |
| hdisk16:117 | lv01:9:2 | stale |
| hdisk16:118 | lv01:10:2 | stale |

In der ersten Spalte sind die physischen Partitionen und in der zweiten Spalte die logischen Partitionen aufgelistet. Wenn eine physische Partition veraltet ist, steht in der dritten Spalte der Wert "stale".

Zugehörige Tasks:

„Fehlerbehebung im Zusammenhang mit Plattenlaufwerken“ auf Seite 396

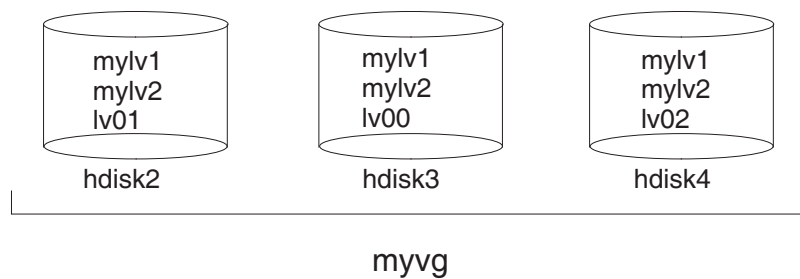
Im Folgenden wird beschrieben, wie Sie Probleme mit Plattenlaufwerken diagnostizieren und beheben.

Wiederherstellung mit einem erneut formatierten oder Ersatzplattenlaufwerk:

Sie können Daten von einem defekten Plattenlaufwerk wiederherstellen, wenn Sie die defekte Platte erneut formatieren oder austauschen müssen.

Achtung: Bevor Sie ein Plattenlaufwerk erneut formatieren oder austauschen, entfernen Sie alle Referenzen auf nicht gespiegelte Dateisysteme von der defekten Platte. Entfernen Sie anschließend die Platte aus der Datenträgergruppe und aus der Systemkonfiguration. Wenn Sie dies nicht tun, treten Probleme in der ODM- (Object Data Manager) und in der Systemkonfigurationsdatenbank auf. Anweisungen zu diesen wichtigen Schritten finden Sie in der folgenden Prozedur unter der Überschrift "Vor dem Austausch bzw. der erneuten Formatierung einer defekten oder ausfallenden Platte".

Für die folgende Prozedur wird ein Szenario verwendet, in dem die Datenträgergruppe *myvg* drei Plattenlaufwerke mit den Namen *hdisk2*, *hdisk3* und *hdisk4* enthält. In diesem Szenario fällt *hdisk3* aus. Der nicht gespiegelte logische Datenträger *lv01* und eine Kopie des logischen Datenträgers *mylv* sind auf *hdisk2* enthalten. Der logische Datenträger *mylv* ist gespiegelt und hat drei Kopien, die jeweils zwei physische Partitionen auf seiner Platte einnehmen. Das defekte Plattenlaufwerk *hdisk3* enthält eine weitere Kopie von *mylv* und den nicht gespiegelten logischen Datenträger *lv00*. *hdisk4* enthält eine dritte Kopie von *mylv* sowie einen logischen Datenträger mit dem Namen *lv02*. Die folgende Abbildung zeigt dieses Szenario.



Diese Prozedur lässt sich in die folgenden Schlüsselsegmente aufteilen:

- Dinge, die vor dem Austausch oder der erneuten Formatierung der ausfallenden Platte zum Schutz der Daten getan werden müssen,
- die für die erneute Formatierung bzw. für den Austausch der Platte zu befolgende Prozedur,
- Dinge, die nach der erneuten Formatierung bzw. nach dem Austausch der Platte getan werden müssen.

Vor dem Austausch bzw. der erneuten Formatierung einer defekten oder ausgefallenen Platte:

1. Melden Sie sich als Benutzer mit Rootberechtigung an.
2. Wenn Sie mit den logischen Datenträgern auf dem ausfallenden Laufwerk nicht vertraut sind, verwenden Sie eine funktionsfähige Platte, um den Inhalt der ausfallenden Platte anzuzeigen. Wenn Sie beispielsweise *hdisk4* zum Anzeigen des Inhalts von *hdisk3* verwenden möchten, geben Sie in der Befehlszeile Folgendes ein:

```
lspv -M -n hdisk4 hdisk3
```

Der Befehl **lspv** zeigt Informationen über einen physischen Datenträger in einer Datenträgergruppe an. Es erscheint eine Ausgabe wie die folgende:

```
hdisk3:1      mylv:1
hdisk3:2      mylv:2
hdisk3:3      lv00:1
hdisk3:4-50
```

In der ersten Spalte sind die physischen Partitionen und in der zweiten Spalte die logischen Partitionen aufgelistet. Die Partitionen 4 bis 50 sind frei.

3. Sichern Sie, sofern möglich, alle logischen Datenträger auf der ausfallenden Einheit, die nur eine Kopie haben. Diesbezügliche Anweisungen finden Sie im Abschnitt „Benutzerdateien und -dateisysteme sichern“ auf Seite 28.
4. Wenn Sie Dateisysteme mit nur einer Kopie haben, hängen Sie sie von der Platte ab. (Dateisysteme mit nur einer Kopie können Sie mithilfe der Ausgabe des Befehls **lspv** ermitteln. In dieser Ausgabe

haben solche Dateisysteme dieselbe Anzahl an logischen Partitionen wie physischen Partitionen.) Gespiegelte Dateisysteme müssen nicht abgehängt werden.

In diesem Szenario ist `lv00` auf der ausfallenden Platte `hdisk3` ein Dateisystem mit nur einer Kopie. Geben Sie Folgendes ein, um das Dateisystem abzuhängen:

```
umount /dev/lv00
```

Wenn Sie den Namen des Dateisystems nicht kennen, geben Sie (vorausgesetzt, dass sich die Datei `/etc/filesystems` nicht nur auf der ausgefallenen Platte befindet) in der Befehlszeile den Befehl `mount` ein, um alle angehängten Dateisysteme aufzulisten und den Namen zu ermitteln, der Ihrem logischen Datenträger zugeordnet ist. Sie können auch den Befehl `grep` verwenden und mit diesem Befehl die Datei `/etc/filesystems` angeben, um nur die Dateisystemnamen aufzulisten, die Ihrem logischen Datenträger zugeordnet sind. Beispiel:

```
grep lv00 /etc/filesystems
```

Die Ausgabe gleicht dem folgenden Beispiel:

```
dev          = /dev/lv00
log          = /dev/loglv00
```

Anmerkungen:

- a. Der Befehl **umount** scheitert, wenn das Dateisystem, das Sie abhängen möchten, derzeit im Gebrauch ist. Der Befehl **umount** wird nur ausgeführt, wenn keine Dateien im Dateisystem geöffnet sind und keine aktuellen Verzeichnisse von Benutzern auf dieser Einheit vorhanden sind.
 - b. Ein anderer Name für den Befehl **umount** ist **umount**. Die Namen sind gegeneinander austauschbar.
5. Entfernen Sie mit dem folgenden Befehl **rmfs** nacheinander alle Dateisysteme mit nur einer Kopie vom ausgefallenen physischen Datenträger:

```
rmfs /Name_des_Dateisystems
```

6. Entfernen Sie alle gespiegelten logischen Datenträger auf der ausfallenden Platte.

Anmerkung: Sie können den Befehl **rmlvcopy** nicht verwenden, um die logischen Datenträger `hd5` und `hd7` von physischen Datenträgern in der Datenträgergruppe `rootvg` zu entfernen. Das System lässt das Entfernen dieser logischen Datenträger nicht zu, weil es jeweils nur eine Kopie dieser Datenträger gibt.

Der Befehl **rmlvcopy** entfernt Kopien aus jeder logischen Partition. Geben Sie beispielsweise Folgendes ein:

```
rmlvcopy mylv 2 hdisk3
```

Indem Sie die Kopie auf `hdisk3` entfernen, reduzieren Sie die Anzahl der Kopien jeder logischen Partition, die zum logischen Datenträger `mylv` gehört, von drei auf zwei (eine auf `hdisk4` und eine auf `hdisk2`).

7. Wenn die ausfallende Platte zur Stammdatenträgergruppe gehört und den logischen Datenträger `hd7` enthält, entfernen Sie die primäre Speicherauszugseinheit (`hd7`), indem Sie in der Befehlszeile Folgendes eingeben:

```
sysdumpdev -P -p /dev/sysdumpnull
```

Der Befehl **sysdumpdev** ändert die Position der primären bzw. sekundären Speicherauszugseinheit für ein laufendes System. Bei einem Warmstart wird die Speicherauszugseinheit an die ursprüngliche Position zurückversetzt.

Anmerkung: Sie können auch eine DVD als Speicherauszugseinheit auswählen. Weitere Informationen zum Konfigurieren einer DVD als Speicherauszugseinheit finden Sie unter **sysdumpdev**.

8. Entfernen Sie mit dem folgenden Befehl alle Paging-Bereiche auf der Platte:

```
rmpps PB-Name
```


PB-Name steht für den Namen des zu entfernenden Paging-Bereichs, d. h. eigentlich für den Namen des logischen Datenträgers, auf dem sich der Paging-Bereich befindet.

Wenn der Befehl **rmpps** nicht erfolgreich ist, müssen Sie den Direktaufruf **smit chps** verwenden, um den primären Speicherbereich zu inaktivieren und einen Warmstart durchzuführen, bevor Sie mit dieser Prozedur fortfahren. Der Befehl **reducevg** in Schritt 10 kann scheitern, wenn aktive Paging-Bereiche vorhanden sind.

9. Entfernen Sie mit dem Befehl **rmlv** alle weiteren logischen Datenträger aus der Datenträgergruppe. Geben Sie beispielsweise Folgendes ein:

```
rmlv -f lv00
```

10. Entfernen Sie mit dem Befehl **reducevg** die ausgefallene Platte aus der Datenträgergruppe. Geben Sie beispielsweise Folgendes ein:

```
reducevg -df myvg hdisk3
```

Wenn Sie den Befehl **reducevg** nicht ausführen können oder der Befehl nicht erfolgreich ist, können Sie mit der Prozedur in Schritt 13 die VGDA/ODM-Informationen löschen, nachdem Sie das Laufwerk erneut formatiert oder ausgetauscht haben.

Gehen Sie zum Austausch bzw. zur erneuten Formatierung der ausgefallenen oder defekten Platte wie folgt vor:

11. Der nächste Schritt richtet sich danach, ob Sie die Platte erneut formatieren oder austauschen möchten und welchen Hardwaretyp Sie verwenden:
 - Verwenden Sie die folgende Prozedur, wenn Sie das Plattenlaufwerk erneut formatieren möchten:
 - a. Melden Sie sich als Benutzer mit Root-Berechtigung an, und geben Sie in der Befehlszeile den folgenden SMIT-Direktaufruf ein:

```
smit diag
```
 - b. Wählen Sie **Diagnoseprogramme in der aktuellen Shell** aus, um das Diagnosetool von AIX aufzurufen.
 - c. Drücken Sie die Eingabetaste, nachdem Sie die Anzeige **Diagnose-Operations-Anweisungen** gelesen haben.
 - d. Wählen Sie **Taskauswahl** aus.
 - e. Blättern Sie die Taskliste durch, bis Sie die Option **Datenträgerformatierung** finden, und wählen Sie diese Option dann aus.
 - f. Wählen Sie die Platte aus, die Sie erneut formatieren möchten. Nachdem Sie die erneute Formatierung der Platte bestätigt haben, wird der gesamte Inhalt der Platte gelöscht.

Fahren Sie nach Abschluss der erneuten Formatierung der Platte mit Schritt 12 fort.

- Wenn Ihr System Hot-Swap-Platten unterstützt, verwenden Sie die Prozedur im Abschnitt „Wiederherstellung nach einem Plattenausfall, während das System verfügbar bleibt“ auf Seite 404, und fahren Sie anschließend mit Schritt 13 fort.
- Wenn Ihr System keine Hot-Swap-Platten unterstützt, gehen Sie folgendermaßen vor:
 - Schalten Sie das alte Laufwerk mit dem SMIT-Direktaufruf **smit rmvdsk** aus. Setzen Sie das Feld "Definition in Datenbank speichern" auf **Nein**.
 - Wenden Sie sich für den Austausch des Plattenlaufwerks an Ihre Systemunterstützung.

Gehen Sie nach dem Austausch bzw. der erneuten Formatierung der ausgefallenen oder defekten Platte wie folgt vor:

12. Befolgen Sie die Anweisungen in den Abschnitten „Eine Platte konfigurieren“ auf Seite 386 und „Eine verfügbare Platte zu einem physischen Datenträger machen“ auf Seite 388.
13. Wenn Sie vor der Formatierung der Platte (Schritt 10) den Befehl **reducevg** zum Entfernen der Platte aus der alten Datenträgergruppe nicht verwenden können, können Sie mit der folgenden Prozedur die VGDA/ODM-Informationen löschen.
 - a. Wenn die Datenträgergruppe nur diese eine Platte enthalten hat, die erneut formatiert wurde, geben Sie Folgendes ein:

`exportvg Name_der_Datenträgergruppe`

`Name_der_Datenträgergruppe` steht für den Namen Ihrer Datenträgergruppe.

- b. Wenn die Datenträgergruppe mehrere Platten enthält, geben Sie in der Befehlszeile Folgendes ein:
`varyonvg Name_der_Datenträgergruppe`

Das System zeigt eine Nachricht über eine fehlende oder nicht verfügbare Platte an, und die neue (bzw. erneut formatierte) Platte wird aufgelistet. Notieren Sie die PVID (Physical Volume Identifier, ID des physischen Datenträgers) der neuen Platte, die in der **varyonvg**-Nachricht aufgeführt ist. Es handelt sich dabei um die 16-stellige Zeichenfolge, die zwischen dem Namen der fehlenden Platte und dem Kennsatz PVNOTFND angegeben ist. Beispiel:

```
hdisk3 00083772caa7896e PVNOTFND
```

Geben Sie Folgendes ein:

```
varyonvg -f Name_der_Datenträgergruppe
```

Die fehlende Platte wird jetzt mit dem Kennsatz PVREMOVED angezeigt. Beispiel:

```
hdisk3 00083772caa7896e PVREMOVED
```

Geben Sie anschließend den folgenden Befehl ein:

```
reducevg -df Name_der_Datenträgergruppe PVID
```

PVID steht für die ID des physischen Datenträgers (in diesem Szenario 00083772caa7896e).

14. Verwenden Sie den Befehl **extendvg**, um das neue Plattenlaufwerk zur Datenträgergruppe hinzuzufügen. Geben Sie beispielsweise Folgendes ein:

```
extendvg myvg hdisk3
```

15. Verwenden Sie den Befehl **mklv**, um die logischen Datenträger mit nur einer Kopie auf dem neuen (bzw. erneut formatierten) Plattenlaufwerk erneut zu erstellen. Geben Sie beispielsweise Folgendes ein:

```
mklv -y lv00 myvg 1 hdisk3
```

Dieser Beispielbefehl erstellt den logischen Datenträger lv00 auf dem Laufwerk *hdisk3* neu. 1 bedeutet, dass dieser logische Datenträger nicht gespiegelt ist.

16. Verwenden Sie den Befehl **crfs**, um die Dateisysteme auf dem logischen Datenträger erneut zu erstellen. Geben Sie beispielsweise Folgendes ein:

```
crfs -v jfs -d lv00 -m /dev/lv00
```

17. Informationen zum Wiederherstellen von Dateisystemdaten mit nur einer Kopie von einem Sicherungsdatenträger finden Sie im Abschnitt „Benutzerdateien aus einem Sicherungsbild wiederherstellen“ auf Seite 33.

18. Verwenden Sie den Befehl **mklvcopy**, um die gespiegelten Kopien logischer Datenträger erneut zu erstellen. Geben Sie beispielsweise Folgendes ein:

```
mklvcopy mylv 3 hdisk3
```

Dieser Beispielbefehl erstellt eine gespiegelte dritte Partition des logischen Datenträgers *mylv* auf *hdisk3*.

19. Verwenden Sie den Befehl **syncvg**, um die neue Spiegelung mit den Daten in den anderen Spiegelungen (in diesem Beispiel *hdisk2* und *hdisk4*) zu synchronisieren. Geben Sie beispielsweise Folgendes ein:

```
syncvg -p hdisk3
```

Alle gespiegelten Dateisysteme müssen wiederhergestellt und auf dem aktuellen Stand sein. Wenn Sie Ihre Dateisysteme mit nur einer Kopie sichern konnten, können auch diese wieder verwendet werden. Sie sollten jetzt in der Lage sein, den normalen Systembetrieb fortzusetzen.

Zugehörige Tasks:

„Fehlerbehebung im Zusammenhang mit Plattenlaufwerken“ auf Seite 396

Im Folgenden wird beschrieben, wie Sie Probleme mit Plattenlaufwerken diagnostizieren und beheben.

Beispiel für die Wiederherstellung nach Ausfall eines Plattenlaufwerks:

Zur Wiederherstellung nach dem Ausfall eines Plattenlaufwerks müssen Sie die Schritte, die Sie zum Erstellen der Datenträgergruppe ausgeführt haben, in umgekehrter Reihenfolge ausführen.

Das folgende Beispiel veranschaulicht dieses Verfahren. Es zeigt Schritt für Schritt, wie ein gespiegelter logischer Datenträger erstellt, geändert und nach einem Plattenausfall wiederhergestellt wird.

Anmerkung: Im folgenden Beispiel wird eine bestimmte Instanz verwendet. Es ist nicht als allgemeiner Prototyp gedacht, den Sie für alle allgemeinen Wiederherstellungsprozeduren verwenden können.

1. Die Systemmanagerin Jane erstellt mit dem folgenden Befehl eine Datenträgergruppe mit dem Namen **workvg** auf hdisk1:

```
mkvg -y workvg hdisk1
```

2. Anschließend erstellt sie mit dem folgenden Befehl zwei weitere Platten für diese Datenträgergruppe:

```
extendvg workvg hdisk2
```

```
extendvg workvg hdisk3
```

3. Jane erstellt einen logischen Datenträger mit einer Größe von 40 MB mit drei Kopien. Jede Kopie befindet sich auf jeweils einer der drei Platten, aus denen sich die Datenträgergruppe **workvg** zusammensetzt. Sie verwendet dazu die folgenden Befehle:

```
mklv -y testlv workvg 10
```

```
mklvcopy testlv 3
```

Nachdem Jane die gespiegelte Datenträgergruppe **workvg** erstellt hat, fällt hdisk2 aus. Sie führt zur Wiederherstellung die folgenden Schritte aus:

1. Sie entfernt die Kopie des logischen Datenträgers mit dem folgenden Befehl von hdisk2:

```
rmlvcopy testlv 2 hdisk2
```

2. Sie hängt hdisk2 mit dem folgenden Befehl vom System ab, so dass der ODM und der VGDA aktualisiert werden:

```
reducevg workvg hdisk2
```

3. Sie entfernt hdisk2 mit dem folgenden Befehl aus der Systemkonfiguration, um den Austausch vorzubereiten:

```
rmdev -l hdisk2 -d
```

4. Sie fährt das System mit dem folgenden Befehl herunter:

```
shutdown -F
```

5. Sie tauscht die Platte aus. Die neue Platte hat nicht dieselbe SCSI-ID wie die vorherige hdisk2.

6. Sie führt einen Warmstart durch.

Da Sie eine neue Platte haben (das System erkennt, dass eine neue PVID auf der Platte vorhanden ist), wählt das System den ersten *freien* hdisk-Namen. Da in Schritt 3 das Flag **-d** verwendet und der Name hdisk2 freigegeben wurde, wählt das System hdisk2 als Namen für die neue Platte aus. Wenn das Flag **-d** nicht verwendet worden wäre, würde hdisk4 als neuer Name ausgewählt werden.

7. Jane fügt die Platte mit dem folgenden Befehl zur Datenträgergruppe **workvg** hinzu:

```
extendvg workvg hdisk2
```

8. Sie erstellt mit dem folgenden Befehl zwei gespiegelte Kopien des logischen Datenträgers:

```
mklvcopy testlv 3
```

Der Logical Volume Manager kopiert die dritte Kopie des logischen Datenträgers automatisch auf die neue Platte hdisk2.

Wiederherstellung nach einem Plattenausfall, während das System verfügbar bleibt:

Die Wiederherstellung nach einem Plattenausfall ist mit der Funktion für das Entfernen von Einheiten bei laufendem Betrieb möglich.

Die Prozedur für die Wiederherstellung nach einem Plattenausfall mit der Funktion für das Entfernen von Einheiten bei laufendem Betrieb ist zum größten Teil identisch mit der Prozedur, die im Abschnitt „Wiederherstellung von Plattenlaufwerken ohne erneute Formatierung“ auf Seite 398 beschrieben ist. Die folgenden Ausnahmen sind zu beachten:

1. Verwenden Sie zum Abhängen von Dateisystemen auf einer Platte die Prozedur "Ein JFS oder JFS2 anhängen".
2. Verwenden Sie zum Entfernen der Platte aus der Datenträgergruppe und aus dem Betriebssystem die Prozedur „Eine Platte ohne Daten entfernen“ auf Seite 424.
3. Sie müssen das System nicht herunterfahren, um die ausgefallene Platte durch eine neue zu ersetzen. Verwenden Sie die folgenden Prozeduren in der angegebenen Reihenfolge:
 - a. „Speicherung logischer Datenträger“ auf Seite 409
 - b. „Eine Platte konfigurieren“ auf Seite 386
 - c. Fahren Sie mit Schritt 13 der Prozedur „Wiederherstellung mit einem erneut formatierten oder Ersatzplattenlaufwerk“ auf Seite 398 fort.

Zugehörige Tasks:

„Fehlerbehebung im Zusammenhang mit Plattenlaufwerken“ auf Seite 396

Im Folgenden wird beschrieben, wie Sie Probleme mit Plattenlaufwerken diagnostizieren und beheben.

Eine Platte in einer Datenträgergruppe mit nur einer Platte austauschen:

Verwenden Sie eine der folgenden Prozeduren, wenn Sie auf eine Platte zugreifen können, die in einer Datenträgergruppe nicht mehr verwendet werden kann.

- „Inhalt eines physischen Datenträgers migrieren“ auf Seite 384

Wenn die Platte defekt und nicht mehr zugänglich ist, führen Sie die folgenden Schritte aus:

1. Exportieren Sie die Datenträgergruppe.
2. Tauschen Sie das Laufwerk aus.
3. Stellen Sie die Daten vom vorhandenen Sicherungsdaträger wieder her.

Fehler bei physischen und logischen Datenträgern:

Es gibt mehrere häufig auftretende Fehler bei physischen und logischen Datenträgern, die Sie beheben können.

Hot-Spot-Probleme:

Wenn Sie beim Zugriff auf logische Datenträger Leistungseinbußen feststellen, haben Sie möglicherweise so genannte Hot-Spots (Brennpunkte) auf Ihren logischen Datenträgern, die zu viel Platten-E/A verzeichnen.

Weitere Informationen finden Sie im Artikel „Hot-Spot-Verwaltung in logischen Datenträgern“ auf Seite 440.

LVCB-Warnungen:

Eine Warnung wird ausgegeben, wenn der LVCB (Logical Volume Control Block, Steuerblock des logischen Datenträgers) ungültige Informationen enthält.

Der Steuerblock des logischen Datenträgers ist der erste Block eines logischen Datenträgers. Die Größe des LVCB entspricht der Blockgröße der physischen Datenträger in der Datenträgergruppe. Dieser Bereich enthält wichtige Informationen, wie z. B. das Erstellungsdatum des logischen Datenträgers, Informationen zu gespiegelten Kopien und gültige Mountpunkte im JFS. Für die Aktualisierung des LVCB müssen für die Algorithmen des LVM bestimmte LVM-Befehle verwendet werden. Der alte LVCB wird gelesen und analysiert, um festzustellen, ob er gültig ist. Wenn die LVCB-Informationen gültig sind, wird der LVCB aktualisiert. Sind die Informationen nicht gültig, wird der LVCB nicht aktualisiert, und es kann die folgende Nachricht ausgegeben werden:

```
Warning, cannot write lv control block data.
```

Diese Nachricht wird meistens ausgegeben, wenn Datenbankprogramme das JFS umgehen und direkt auf logische Datenträger zugreifen. In diesem Fall werden die Informationen für die Datenbank unverändert über den LVCB geschrieben. Bei logischen Datenträgern ohne Dateisystem hat dies keine schwerwiegenden Folgen. Nach dem Überschreiben des LVCB kann der Benutzer trotzdem die folgenden Aktionen ausführen:

- einen logischen Datenträger erweitern,
- gespiegelte Kopien des logischen Datenträgers erstellen,
- den logischen Datenträger entfernen,
- ein Journaled File System zum Anhängen des logischen Datenträgers erstellen.

Beim Löschen von LVCBs sind bestimmte Einschränkungen zu beachten. Ein logischer Datenträger mit einem gelöschten LVCB kann möglicherweise nicht ordnungsgemäß auf anderen Systemen importiert werden. Während eines Imports überprüft der LVM-Befehl **importvg** die LVCBs aller definierten logischen Datenträger in einer Datenträgergruppe nach Informationen, die sich auf die logischen Datenträger beziehen. Wenn der LVCB nicht vorhanden ist, definiert die importierte Datenträgergruppe trotzdem den logischen Datenträger für das neue System, das auf diese Datenträgergruppe zugreift, und der Benutzer kann trotzdem auf den logischen Datenträger ohne Dateisystem zugreifen. In der Regel geschieht jedoch Folgendes:

- Alle JFS-Informationen gehen verloren, und der zugehörige Mountpunkt wird nicht auf das neue System importiert. In diesem Fall müssen Sie neue Mountpunkte erstellen, und die Verfügbarkeit der zuvor im Dateisystem gespeicherten Daten ist nicht gewährleistet.
- Einige Nicht-JFS-Informationen, die den logischen Datenträger betreffen, werden nicht gefunden. In diesem Fall verwendet das System Standardinformationen für den logischen Datenträger, um die ODM-Informationen bereitzustellen. Deshalb kann ein Teil der Ausgabe des Befehls **lslv** mit dem echten logischen Datenträger nicht konsistent sein. Wenn noch Kopien des logischen Datenträgers auf den ursprünglichen Platten vorhanden sind, werden die Informationen in der ODM-Datenbank nicht korrekt widerspiegelt. Verwenden Sie die Befehle **rmilvcopy** und **mkilvcopy**, um alle Kopien des logischen Datenträgers erneut zu erstellen und den ODM zu synchronisieren.

Grenzwerte für physische Partitionen:

Im Design des Logical Volume Manager (LVM) ist jede logische Partition einer physischen Partition (PP) zugeordnet. Jeder physischen Partition ist eine bestimmte Anzahl von Plattensektoren zugeordnet. Das Design von LVM begrenzt die Anzahl der physischen Partitionen, die LVM pro Platte überwachen kann, auf 1016. In den meisten Fällen werden nicht alle der 1016 zu überwachenden Partitionen von einer Platte verwendet.

Wenn dieser Grenzwert überschritten wird, kann eine Nachricht wie die folgende ausgegeben werden:

```
0516-1162 extdvg: Achtung: Die Größe der physischen Partition von PP-Größe erfordert die Erstellung von Gesamtanzahl_PPs Partitionen für PD-Name. Die Begrenzung für Datenträgergruppe DTG-Name liegt bei Grenzwert physischen Partitionen je physischem Datenträger. Über den Befehl chvg mit der Option -t versuchen, die maximale Anzahl der physischen Partitionen je physischem Datenträger für diese Datenträgergruppe zu ändern.
```

Erläuterung:

PP-Größe

1 MB bis 1 GB in 2er-Potenzen

Gesamtanzahl_PPs

Die Gesamtanzahl der physischen Partitionen auf dieser Platte, berechnet auf der Basis von *PP-Größe*.

PD-Name

Der Name des physischen Datenträgers, z. B. hdisk3.

DTG-Name

Der Name der Datenträgergruppe.

Grenzwert

1016 oder ein Vielfaches von 1016.

Diese Einschränkung kommt in den folgenden Fällen zum Tragen:

1. Beim Erstellen einer Datenträgergruppe mit dem Befehl **mkvg** haben Sie für die Anzahl physischer Partitionen auf einer Platte in der Datenträgergruppe einen Wert größer als 1016 angegeben. Zur Umgehung dieser Einschränkung können Sie zwischen den Größen 1, 2, 4 (Standardwert), 8, 16, 32, 64, 128, 256, 512 und 1024 MB für die physischen Partitionen wählen und den Befehl **mkvg -s** verwenden, um die Datenträgergruppe zu erstellen. Alternativ können Sie einen entsprechenden Faktor verwenden, der ein Vielfaches von 1016 Partitionen pro Platte zulässt, und den Befehl **mkvg -t** verwenden, um die Datenträgergruppe zu erstellen.
2. Wenn Sie eine Platte mit dem Befehl **extendvg** zu einer bereits vorhandenen Datenträgergruppe hinzufügen, wird mit der neuen Platte die Einschränkung von 1016 Partitionen überschritten. Zur Behebung dieses Problems konvertieren Sie die vorhandene Datenträgergruppe mit dem Befehl **chvg -t**, dass sie ein Vielfaches von 1016 Partitionen pro Platte aufnehmen kann. Alternativ können Sie die Datenträgergruppe erneut mit einer größeren Partitionsgröße erstellen, die das Hinzufügen der neuen Platte zulässt, oder Sie können eine eigenständige Datenträgergruppe mit einer größeren Partitionsgröße für die neue Platte erstellen.

Partitionseinschränkungen und Stammdatenträgergruppe rootvg

Wenn der Installationscode feststellt, dass das rootvg-Laufwerk größer ist als 4 GB, ändert er den Wert für **mkvg-s** so lange, bis die gesamte Festplattenkapazität den verfügbaren 1016 Spuren zugeordnet werden kann. Diese Installationsänderung impliziert auch, dass alle anderen Platten, die rootvg hinzugefügt werden, unabhängig von der Größe ebenfalls mit dieser physischen Partitionsgröße definiert werden.

Partitionseinschränkungen und RAID-Systeme

Für Systeme, die eine RAID (Redundant Array of Identical Disks) verwenden, kann der vom LVM verwendete Name `/dev/hdiskX` viele Platten enthalten, die keine 4-GB-Platten sind. In diesem Fall bleibt die Anforderung mit den 1016 Partitionen trotzdem bestehen. Dem LVM ist die Größe der einzelnen Platten, aus denen sich `/dev/hdiskX` zusammensetzt, nicht bekannt. Der LVM basiert die Einschränkung auf 1016 Partitionen auf die bekannte Größe von `/dev/hdiskX` und nicht auf die eigentlichen physischen Platten, aus denen sich `/dev/hdiskX` zusammensetzt.

Synchronisation der Einheitenkonfigurationsdatenbank:

Eine Systemstörung kann dazu führen, dass die Einheitenkonfigurationsdatenbank nicht mehr mit dem LVM konsistent ist. Sie können die Einheitenkonfigurationsdatenbank mit den LVM-Daten synchronisieren.

Wenn die Einheitenkonfigurationsdatenbank nicht mehr mit dem LVM konsistent ist, generiert ein Befehl für logische Datenträger Fehlermeldungen wie die folgenden:

0516-322 Die Einheitenkonfigurationsdatenbank ist inkonsistent.

ODER

0516-306 Logischer Datenträger *Name* nicht in Einheitenkonfigurationsdatenbank gefunden.

(Normalerweise ist der logische Datenträger *Name* verfügbar.)

Achtung: Entfernen Sie keine /dev-Einträge für Datenträgergruppen oder logische Datenträger. Ändern Sie die Datenbankeinträge für Datenträgergruppen oder logische Datenträger nicht mit dem Object Data Manager.

Wenn Sie die Einheitenkonfigurationsdatenbank mit den LVM-Daten synchronisieren möchten, geben Sie in der Befehlszeile den folgenden Befehl ein, für den Sie Rootberechtigung benötigen:

```
synclvodm -v VG-Name
```

VG-Name steht für den Namen der Datenträgergruppe, die Sie synchronisieren möchten.

Zugehörige Informationen:

„Fehler bei Datenträgergruppen beheben“

Verwenden Sie die folgenden Methoden, um Fehler bei Datenträgergruppen zu beheben.

Fehler bei Datenträgergruppen beheben:

Verwenden Sie die folgenden Methoden, um Fehler bei Datenträgergruppen zu beheben.

Wenn der Befehl **importvg** nicht ordnungsgemäß funktioniert, versuchen Sie, die Einheitenkonfigurationsdatenbank zu aktualisieren.

Fehler beim Aktivieren beheben

Achtung: Das Beheben eines Fehlers beim Aktivieren einer Datenträgergruppe (Vary-On) ist eine ungewöhnliche Operation. Überprüfen Sie alle anderen möglichen Fehlerquellen, z. B. Hardware, Kabel, Adapter und Versorgungsstromkreise, bevor Sie die Prozedur fortsetzen. Die Behebung eines Quorumfehlers während der Aktivierung einer Datenträgergruppe ist nur in Notfällen und als letzte Abhilfemaßnahme in Erwägung zu ziehen (z. B. um Daten von einer defekten Platte zu retten). Die Datenintegrität der in den Kopien des VGDA und des VGSA enthaltenen Verwaltungsdaten ist nicht gewährleistet, wenn ein Quorumfehler behoben wird.

Wenn Sie das Aktivieren einer Datenträgergruppe erzwingen, indem Sie das Fehlen eines Quorums überschreiben, wird das Attribut PV STATE (Status des physischen Datenträgers) aller physischer Datenträger, die fehlen, in "removed" (entfernt) geändert. Dies bedeutet, dass alle VGDA- und VGSA-Kopien von diesen physischen Datenträgern entfernt werden. Danach nehmen diese physischen Datenträger nicht mehr an der Quorumprüfung teil. Sie können auch erst dann wieder in der Datenträgergruppe aktiviert werden, wenn Sie sie wieder in die Datenträgergruppe aufnehmen. Das Flag "varyonvg -f" (das für das Überschreiben des Quorumverlusts verwendet wird) wird ignoriert, wenn die Datenträgergruppe das Quorum nicht verloren hat.

In den folgenden Fällen kann der Fehler beim Aktivieren der Datenträgergruppe behoben werden, so dass auf die Daten auf den verfügbaren Platten in der Datenträgergruppe zugegriffen werden kann:

- Nicht verfügbare physische Datenträger scheinen permanent beschädigt zu sein.
- Sie können bestätigen, dass mindestens eine der derzeit zugänglichen physischen Datenträger (die außerdem eine gültige VGDA- und VGSA-Kopie enthalten müssen) beim letzten Aktivieren der Datenträgergruppe online war. Dekonfigurieren und schalten Sie die fehlenden physischen Datenträger so lange aus, bis sie diagnostiziert und repariert werden können.

Verwenden Sie die folgende Prozedur, um den Quorumverlust zu verhindern, wenn eine Platte fehlt oder in Kürze ausfallen könnte und repariert werden muss:

1. Geben Sie Folgendes ein, um den Datenträger vorübergehend aus der Datenträgergruppe zu entfernen:

```
chpv -vr PVname
```

Nach der Ausführung dieses Befehls wird der physische Datenträger *PVname* nicht mehr bei der Quorumprüfung berücksichtigt. In einer Datenträgergruppe mit zwei Platten scheitert dieser Befehl jedoch, wenn Sie versuchen, den Befehl **chpv** für die Platte auszuführen, die die beiden VGDA's/VGSA's enthält. Der Befehl lässt nicht zu, dass das Quorum verloren geht.

2. Wenn Sie die Platte für eine Reparatur entfernen müssen, schalten Sie das System aus, und entfernen Sie die Platte. (Diesbezügliche Anweisungen finden Sie im Abschnitt „Fehlerbehebung im Zusammenhang mit Plattenlaufwerken“ auf Seite 396.) Fahren Sie nach der Reparatur der Platte und dem Wiedereinbau der Platte in das System mit dem nächsten Schritt fort.
3. Geben Sie Folgendes ein, um die Platte wieder in der Datenträgergruppe verfügbar zu machen, damit sie bei der Quorumprüfung berücksichtigt wird:

```
chpv -v a PV-Name
```

Anmerkung: Der Befehl **chpv** wird nur für die Änderung der Quorumprüfung verwendet. Die Daten, die sich auf der Platte befinden, sind weiterhin vorhanden und müssen auf andere Platten verschoben oder kopiert werden, wenn die Platte nicht wieder im System eingebaut wird.

VGDA-Warnungen

In manchen Fällen können beim Hinzufügen einer neuen Platte zu einer vorhandenen Datenträgergruppe oder beim Erstellen einer neuen Datenträgergruppe Probleme auftreten. Der LVM gibt die folgende Nachricht aus:

```
0516-1163 extendlvg: VG-Name verfügt bereits über die maximale Anzahl physischer Datenträger. Bei einer maximalen Anzahl physischer Partitionen je physischem Datenträger von Grenzwert ist die maximale Anzahl der physischen Datenträger für die Datenträgergruppe VG-Name MaxAnzPlatten.
```

Erläuterung:

VG-Name

Ist der Name der Datenträgergruppe.

Grenzwert

Ist 1016 oder ein Vielfaches von 1016.

MaxAnzPlatten

Ist die maximale Anzahl der Platten in einer Datenträgergruppe. Wenn es beispielsweise 1016 physische Partitionen (PPs) pro Platte gibt, entspricht *MaxAnzPlatten* dem Wert 32. Sind 2032 physische Partitionen vorhanden, hat *MaxAnzPlatten* den Wert 16.

Sie können die Datei *image.data* ändern und anschließend eine Installation auf einer alternativen Platte durchführen oder das System mit dem Befehl **mksysb** wiederherstellen, um die Datenträgergruppe erneut als große Datenträgergruppe zu erstellen. Weitere Informationen hierzu finden Sie in *Installation and Migration*.

In früheren Versionen von AIX, in denen der Grenzwert für die Platten kleiner ist als 32, gilt diese Beschreibung der maximalen Anzahl VGDA's nicht für **rootvg**. Um in diesen AIX-Versionen Benutzern beim Erstellen von **rootvg** mehr freien Plattenspeicherplatz zur Verfügung zu stellen, verwendet der Befehl **mkvg -d** die im Installationsmenü ausgewählte Anzahl der Platten als Referenzzahl. Die Zahl, die mit **-d** angegeben wird, ist 7 für eine Platte und erhöht sich mit jeder weiteren ausgewählten Platte um jeweils eins. Werden beispielsweise zwei Platten ausgewählt, ist die Zahl 8, werden drei Platten ausgewählt, ist die Zahl 9 usw.

Zugehörige Konzepte:

„Synchronisation der Einheitenkonfigurationsdatenbank“ auf Seite 406

Eine Systemstörung kann dazu führen, dass die Einheitenkonfigurationsdatenbank nicht mehr mit dem LVM konsistent ist. Sie können die Einheitenkonfigurationsdatenbank mit den LVM-Daten synchronisieren.

Speicherung logischer Datenträger

Logische Datenträger sind Gruppen von Informationen auf physischen Datenträgern.

Für die Verwaltung des Plattenspeichers wird eine Hierarchie von Strukturen verwendet. Jedes einzelne Plattenlaufwerk ist ein so genannter *physischer Datenträger* (PV, Physical Volume) und hat einen Namen, wie z. B. `/dev/hdisk0`. Jeder verwendete physische Datenträger gehört zu einer *Datenträgergruppe* (VG, Volume Group). Alle physischen Datenträger in einer Datenträgergruppe werden in *physische Partitionen* (PP, Physical Partition) identischer Größe aufgeteilt. Für die Plattenplatzreservierung (Bereichszuordnung) wird jeder physische Datenträger in fünf Bereiche eingeteilt: **äußerer Rand**, **innerer Rand**, **äußere Mitte**, **innere Mitte** und **Zentrum**. Die Anzahl physischer Partitionen in jedem Bereich variiert je nach Gesamtkapazität des Plattenlaufwerks.

In jeder Datenträgergruppe wird mindestens ein *logischer Datenträger* (LV, Logical Volume) definiert. Die Daten auf logischen Datenträgern scheinen für den Benutzer zusammenhängend zu sein, können aber auf dem physischen Datenträger unzusammenhängend sein. Auf diese Weise können Dateisysteme, Paging-Bereich und andere logische Datenträger in der Größe verändert oder verlagert werden und sich auf mehrere physische Datenträger erstrecken. Außerdem kann ihr Inhalt repliziert werden, um eine größere Flexibilität und Verfügbarkeit zu erreichen.

Jeder logische Datenträger setzt sich aus mindestens einer *logischen Partition* (LP, Logical Partition) zusammen. Jede logische Partition entspricht mindestens einer physischen Partition. Wenn für den logischen Datenträger Spiegelung angegeben ist, werden weitere physische Partitionen zugeordnet, auf denen die zusätzlichen Kopien jeder logischen Partition gespeichert werden. Obwohl die logischen Partitionen fortlaufend nummeriert werden, sind die zugrunde liegenden physischen Partitionen nicht unbedingt fortlaufend bzw. zusammenhängend.

Logische Datenträger können für zahlreiche Systemzwecke verwendet werden, z. B. für das Paging. Allerdings dient jeder logische Datenträger nur einem einzigen Zweck. Viele logische Datenträger enthalten ein einziges Journalized File System (JFS oder JFS2). Jedes JFS setzt sich aus einem Pool von Blöcken zusammen, die je eine Seite (4 KB) groß sind. Wenn Daten in eine Datei geschrieben werden müssen, werden dieser Datei zusätzliche Blöcke zugeordnet. Diese Blöcke sind möglicherweise nicht miteinander oder anderen Blöcken, die der Datei zuvor zugeordnet wurden, benachbart. Ein Dateisystem kann mit einer Fragmentgröße von weniger als 4 KB (512 Bytes, 1 KB, 2 KB) definiert werden.

Nach der Installation hat das System eine Datenträgergruppe (die Stammdatenträgergruppe (`rootvg`)), die sich aus einer Basisgruppe logischer Datenträger, die für das Starten des Systems erforderlich sind, und allen weiteren logischen Datenträgern zusammensetzt, die Sie mit dem Installationsscript angeben. Alle anderen physischen Datenträger, die Sie an das System angeschlossen haben, können einer Datenträgergruppe (mit dem Befehl `extendvg`) zugeordnet werden. Sie können den physischen Datenträger der Datenträgergruppe `rootvg` oder einer anderen Datenträgergruppe (die mit dem Befehl `mkvg` definiert wird) hinzufügen. Logische Datenträger können mit den Befehlen oder der menügeführten Schnittstelle SMIT (System Management Interface Tool) angepasst werden.

Zugehörige Tasks:

„Inhalt eines physischen Datenträgers migrieren“ auf Seite 384

„Administrator beim Fehlen eines physischen Datenträgers benachrichtigen“ auf Seite 393

Obwohl AIX einen Fehler protokolliert, wenn ein physischer Datenträger nicht mehr zugänglich ist, kann ein Fehler in bestimmten Situationen unerkannt bleiben.

„Einen gespiegelten Datenträger von einer Datenträgergruppe abtrennen“ auf Seite 395

Sie können durch *Momentaufnahmen* (oder Snapshots) die Konsistenz gespiegelter Datenträgergruppen

schützen und damit potenziellen Plattenausfällen vorbeugen.

„Größe eines Dateisystems in der Stammdatenträgergruppe verringern“ auf Seite 472

Sie können alle *Dateisysteme* ganz einfach auf ihre Mindestgröße verkleinern, indem Sie beim Wiederherstellen des Basisbetriebssystems von der Sicherung die Option **Verkleinern** auf **Ja** setzen.

Installation einer Einheit vorbereiten

Die Installation von Einheiten im System setzt sich aus den folgenden Schritten zusammen: Anschlussposition für die Einheit ermitteln, Einheit physisch anschließen, Einheit mit dem Konfigurationsmanager oder SMIT konfigurieren.

Anmerkung: Für die folgende Prozedur muss das System zur Installation der Einheit heruntergefahren werden. Nicht alle Einheiteninstallationen erfordern einen Systemabschluss. Nähere Informationen hierzu können Sie der Dokumentation entnehmen, die zur jeweiligen Einheit geliefert wurde.

Dieser Abschnitt beschreibt die Installationstasks, die für alle Einheiten dieselben sind. Da Sie in Ihrem System eine Vielzahl von Einheiten installieren können, ist die beschriebene Prozedur allgemein gehalten. Spezielle Informationen finden Sie in den Installationsanweisungen, die Sie zur jeweiligen Einheit erhalten haben.

1. Stoppen Sie alle Anwendungen, die auf der Systemeinheit ausgeführt werden, und fahren Sie die Systemeinheit anschließend mit dem Befehl **shutdown** herunter.
2. Schalten Sie die Systemeinheit und alle angeschlossenen Einheiten aus.
3. Ziehen Sie den Netzstecker der Systemeinheit und aller angeschlossenen Einheiten.
4. Schließen Sie die neue Einheit gemäß der im Installations- und Bedienerhandbuch für die Einheit beschriebenen Prozedur an das System an.
5. Stecken Sie die Netzstecker der Systemeinheit und aller angeschlossenen Einheiten wieder ein.
6. Schalten Sie alle angeschlossenen Einheiten ein, aber lassen Sie das System ausgeschaltet.
7. Schalten Sie die Systemeinheit erst ein, wenn alle Einheiten ihre Selbsttests beim Einschalten (POST, Power-On Self-Tests) abgeschlossen haben.

Der Konfigurationsmanager überprüft automatisch die angeschlossenen Einheiten und konfiguriert alle neuen Einheiten, die er erkennt. Die neuen Einheiten werden mit Standardattributen konfiguriert und in der angepassten Konfigurationsdatenbank mit dem Status **Verfügbar** (Available) protokolliert.

Sie können eine Einheit manuell mit dem SMIT-Direktaufruf **smit dev** konfigurieren. Wenn Sie die Einheitenattribute anpassen müssen oder die Einheit nicht automatisch konfiguriert werden kann, machen Sie sich mithilfe der Dokumentation, die Sie zur Einheit erhalten haben, mit den speziellen Konfigurationsanforderungen vertraut.

Zugehörige Konzepte:

„Einheitenkonfigurationsdatenbank und Einheitenverwaltung“ auf Seite 564

Einheitendaten sind in einer vordefinierten oder angepassten Datenbank enthalten, die als Einheitenkonfigurationsdatenbank verwendet wird.

Laufwerk für wieder beschreibbare optische Platten konfigurieren

Für die Konfiguration eines Laufwerks für wieder beschreibbare optische Platten stehen zwei Methoden zur Verfügung.

Das Laufwerk für wieder beschreibbare optische Platten muss an das System angeschlossen und eingeschaltet sein.

Methode 1

Die Methode 1 ist die schnellere der beiden Methoden. Sie konfiguriert nur das angegebene Laufwerk für wieder beschreibbare optische Platten. Zur Verwendung dieser Methode müssen Sie die folgenden Informationen bereitstellen:

| Eintrag | Beschreibung |
|---------------------------------|--|
| Unterklasse | Definiert, wie das Laufwerk angeschlossen ist. |
| Typ | Gibt den Typ des Laufwerks für wieder beschreibbare optische Platten an. |
| Name der übergeordneten Einheit | Gibt den Systemanschluss an, an den die Einheit angeschlossen ist. |
| Anschlussposition | Gibt die logische Adresse des Laufwerks an. |

Geben Sie den folgenden Befehl ein, um das Laufwerk für wieder beschreibbare optische Platten zu konfigurieren:

```
mkdev -c rwoptical -s Unterklasse -t Typ -p Name_der_übergeordneten_Einheit -w Anschlussposition
```

Im Folgenden sehen Sie ein Beispiel für ein Laufwerk für wieder beschreibbare optische Platten, das die SCSI-ID 6 und die LUN (Logical Unit Number, Nummer der logischen Einheit) null hat und an den dritten SCSI-Bus (scsi3) angeschlossen ist:

```
mkdev -c rwoptical -s scsi -t osomd -p scsi3 -w 6,0 -a pv=yes
```

Methode 2

Methode 2 verwendet den Konfigurationsmanager. Sie durchsucht die aktuelle Konfiguration, erkennt alle neuen Einheiten und konfiguriert die Einheiten automatisch. Diese Methode wird verwendet, wenn nur wenige Informationen über das Laufwerk für wieder beschreibbare optische Platten bekannt sind.

1. Verwenden Sie den Konfigurationsmanager, um alle neu erkannten Einheiten im System (einschließlich des Laufwerks für wieder beschreibbare optische Platten) zu konfigurieren. Geben Sie dazu Folgendes ein:

```
cfgmgr
```

2. Geben Sie den folgenden Befehl ein, um die Namen, die Positionscodes und die Typen aller derzeit konfigurierten Laufwerke für wieder beschreibbare optische Platten aufzulisten:

```
lsdev -C -c rwoptical
```

3. Ermitteln Sie den Namen des neu konfigurierten Laufwerks für wieder beschreibbare optische Platten mit dem Positionscod, der der Position des Laufwerks entspricht, das hinzugefügt werden soll.

Eine große Anzahl von Einheiten konfigurieren

Zu Einheiten gehören Hardwarekomponenten wie Drucker, Laufwerke, Adapter, Busse und Platteneinschübe sowie Pseudoeinheiten wie die Fehlergerätedatei und die Nullgerätedatei. Einheitentreiber sind im Verzeichnis `/usr/lib/drivers` gespeichert.

Die Anzahl der von AIX unterstützten Einheiten kann je nach System in Abhängigkeit von mehreren wichtigen Faktoren variieren. Die folgenden Faktoren können Auswirkungen auf die Dateisysteme haben, die die Einheiten unterstützen:

- Bei der Konfiguration einer großen Anzahl von Einheiten müssen mehr Daten in der ODM-Einheitenkonfigurationsdatenbank gespeichert werden. Außerdem sind möglicherweise mehr Einheitengerätedateien erforderlich. Deshalb werden mehr Plattenspeicherplatz und mehr I-Nodes im Dateisystem benötigt.
- Einige Einheiten erfordern mehr Speicherplatz als andere in der ODM-Einheitenkonfigurationsdatenbank. Auch die Anzahl der verwendeten Gerätedateien oder I-Nodes variiert je nach Einheit. Deshalb richtet sich die benötigte Menge an Speicherplatz und I-Nodes im Dateisystem nach den Typen der Einheiten im System.
- MPIO-Einheiten (Multipath I/O) erfordern mehr Speicherplatz als Nicht-MPIO-Einheiten, da die Daten für die Einheit selbst und für jeden Pfad zur Einheit im ODM gespeichert werden. Als grobe Richtlinie können Sie davon ausgehen, dass für jeden Pfad ein Fünftel des Plattenspeicherplatzes einer Einheit erforderlich ist. Eine MPIO-Einheit mit fünf Pfaden belegt beispielsweise den Speicherplatz von zwei Nicht-MPIO-Einheiten.

- AIX speichert sowohl logische Einheiten als auch physische Einheiten in der ODM-Einheitenkonfigurationsdatenbank. Zu den logischen Einheiten gehören Datenträgergruppen, logische Datenträger, Netzchnittstellen usw. Manchmal kann die Beziehung zwischen logischen und physischen Einheiten die Gesamtanzahl der unterstützten Einheiten erheblich beeinflussen. Wenn Sie beispielsweise eine Datenträgergruppe mit zwei logischen Datenträgern für jede physische Platte definieren, die an ein System angeschlossen ist, ergibt dies vier AIX-Einheiten für jede Platte. Wenn Sie eine Datenträgergruppe mit sechs logischen Datenträgern für jede physische Platte definieren, ergibt dies acht AIX-Einheiten für jede Platte. Deshalb könnten nur halb so viele Platten angeschlossen werden.
- Wenn Sie die Standardeinstellungen der Einheitenattribute ändern, hat dies zur Folge, dass die ODM-Einheitendatenbank größer wird und unter Umständen weniger Einheiten unterstützt werden können.
- Mehr Einheiten erfordern mehr Realspeicher.

AIX verwendet zwei Dateisysteme für die Unterstützung von Einheiten:

- Das RAM-Dateisystem (Random Access Memory, Arbeitsspeicher) wird während des Bootvorgangs in einer Umgebung verwendet, die keinen Paging-Bereich hat und in der keine Plattendateisysteme angehängt sind. Die Größe des RAM-Dateisystems beträgt 25 % des Systemspeichers bis hin zu einem Maximum von 128 MB. Für jedes KB im RAM-Dateisystem wird ein I-Node zugeordnet. Die Mindestanforderungen für den Systemspeicher im Betriebssystem AIX sind 256 MB, was sich in eine Mindestgröße von 64 MB mit 65536 I-Nodes für das Dateisystem des Arbeitsspeichers übersetzen lässt. Wenn der Systemspeicher eine Größe von 512 MB oder mehr hat, hat das RAM-Dateisystem eine maximale Größe von 128 MB mit 131072 I-Nodes. Sollte der benötigte Speicherplatz im RAM-Dateisystem oder die benötigte Anzahl der I-Nodes für die Unterstützung der angeschlossenen Einheiten die für die Platte mit dem Arbeitsspeicher reservierte Menge überschreiten, wird das System möglicherweise nicht gebootet. In diesem Fall müssen Sie einige Einheiten entfernen.
- Der Speicherplatz und die Anzahl der I-Nodes im Stammdateisystem (rootvg) auf der Platte können erhöht werden, solange es nicht zugeordnete Partitionen in rootvg gibt. Mit der maximalen Größe für das Dateisystem des Arbeitsspeichers können aller Voraussicht nach bis zu 25.000 AIX-Geräte konfiguriert werden. Diese Werte beziehen sich auf physische und logische Einheiten. Je nach den verschiedenen Faktoren, die in diesen Abschnitten erwähnt wurden, ist es möglich, dass Sie in Ihrem System mehr oder weniger als die genannten Mengen von Einheiten installieren können.

Anmerkung: Wenn sehr viele Einheiten im System vorhanden sind, kann sich durch die längere Konfigurationszeit eine längere Bootzeit ergeben.

Zugehörige Konzepte:

„Einführung in AIX für BSD-Systemmanager“ auf Seite 342

Die folgenden Tipps sind für BSD-Systemmanager (Berkeley Software Distribution) als Einführung in die Verwaltung von AIX bestimmt.

Laufwerk für austauschbare Datenträger hinzufügen

Sie können ein Laufwerk für austauschbare Datenträger hinzufügen.

In der folgenden Prozedur wird SMIT verwendet, um ein CD-ROM-Laufwerk zu einem System hinzuzufügen. Für andere Arten von Laufwerken für austauschbare Datenträger werden zwar andere Direktaufrufe verwendet, aber die Vorgehensweise ist im Prinzip immer dieselbe. Sie können ein Laufwerk für austauschbare Datenträger auch mit dem Konfigurationsmanager oder mit dem Befehl **mkdev** hinzufügen.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Installieren Sie das CD-ROM-Laufwerk gemäß den Anweisungen in der Dokumentation, die Sie zusammen mit Ihrem System erhalten haben.
2. Melden Sie sich unter einem Benutzernamen mit Rootberechtigung an, und geben Sie den folgenden SMIT-Direktaufruf ein:

```
smit makcdr
```
3. Wählen Sie in der folgenden Anzeige in der Liste der unterstützten Laufwerke den Laufwerktyp aus.

4. Wählen Sie in der folgenden Anzeige in der Liste den übergeordneten Adapter aus.
5. Wählen Sie in der folgenden Anzeige in der Liste mindestens die Verbindungsadresse aus. Sie können in dieser Anzeige auch noch weitere Optionen auswählen. Drücken Sie nach Angabe aller gewünschten Optionen die Eingabetaste. SMIT fügt daraufhin das neue CD-ROM-Laufwerk hinzu.

Jetzt wird das neue CD-ROM-Laufwerk vom System erkannt. Verwenden Sie zum Hinzufügen eines Laufwerks für wieder beschreibbare optische Platten den Direktaufruf **smit makomd**. Verwenden Sie zum Hinzufügen eines Bandlaufwerks den Direktaufruf **smit maktpe**.

Weitere Informationen finden Sie in der Beschreibung des Befehls **mkdev** in der Veröffentlichung *Commands Reference, Volume 3*.

Unterstützung der Speicherplatzrückforderung für logische Datenträger

In AIX 7.2 mit Technology Level 7200-01 und höher unterstützt der Logical Volume Manager (LVM) die Speicherplatzrückforderung für physische Datenträger, die in der Lage sind, Speicherplatz zurückzufordern.

LVM informiert den Plattentreiber, der wiederum das Speichersubsystem darüber informiert, dass der Speicherplatz der Partition nicht mehr im Gebrauch ist, sodass das Speichersubsystem den zugeordneten Speicherplatz zurückfordern kann. Der Plattentreiber hilft dem LVM zu erkennen, ob der physische Datenträger in der Lage ist, Speicherplatz zurückzufordern. Der LVM und die Konfigurationsbefehle für das Dateisystem wie **rmlv**, **rmlvcopy** und **chfs(Shrink fs)** leiten die Speicherplatzrückforderung für die Partitionen ein, nachdem diese freigegeben wurden. Der LVM erkennt, ob der physische Datenträger in der Lage ist, Speicherplatz zurückzufordern, wenn er den Datenträger während der Ausführung des Befehls **varyonvg** oder **extendvg** öffnet. Der LVM versucht auch, dies zu erkennen, während die Datenträgergruppe online ist. Wenn die Statusänderungserkennung ein erneutes Öffnen des physischen Datenträgers erfordert, muss der Administrator den Befehl **varyoffvg** und anschließend den Befehl **varyonvg** für die Datenträgergruppe ausführen.

Datenträgergruppen, die vor AIX 7.2 mit Technology Level 1 erstellt wurden, haben möglicherweise freien Partitionsspeicherplatz, der für eine automatische Rückforderung nicht infrage kommt. Der Administrator kann einen logischen Pseudodatenträger auf diesen freien Partitionen erstellen und löschen, um diesen Speicherplatz zurückzufordern. Speicherplatz wird jedoch für die Partitionen, die nach der Installation von AIX 7.2 mit Technology Level 1 freigegeben werden, automatisch zurückgefordert.

Der LVM-Prozess für die Rückforderung des Speicherplatzes wird nach Abschluss der Ausführung eines Befehls wie **rmlv** im Hintergrund ausgeführt. Wenn das System abstürzt, bevor der LVM-Prozess die Rückforderung des Speicherplatzes für alle Partitionen abgeschlossen hat, werden die Partitionen zwar freigegeben, aber der Speicherplatz für die ausstehenden Partitionen wird nicht zurückgefordert. In einem solchen Szenario können Sie einen logischen Pseudodatenträger erstellen und löschen, um den Speicherplatz von den verbleibenden Partitionen zurückzufordern.

Der LVM-Prozess verzögert die Verarbeitung des Befehls **varyoffvg** oder **reducevg** auch dann nicht, wenn die Speicherplatzrückforderung noch aussteht. Die Speicherplatzrückforderung wird verworfen und es wird nicht auf die Beendigung des Prozesses gewartet.

Anmerkung: Befehle warten nur so lange, bis alle ausstehenden Anforderungen zur Speicherplatzrückforderung an den Plattentreiber übergeben wurden.

Die Funktionalität für die Speicherplatzrückforderung steht im Speichersubsystem zur Verfügung, um den freigegebenen Speicherplatz von einem physischen Datenträger zurückzufordern. Jedes Speichersubsystem erwartet, dass die Rückforderungsanforderung an der jeweiligen Anzahl physischer Blöcke ausgerichtet ist, und die Anzahl physischer Blöcke variiert je nach Speichersubsystem. Deshalb ist es manchmal nicht möglich, Blöcke (alle oder einige) von einer Partition zurückzufordern, wenn die Rückforderungsgröße nicht auf die physischen Blöcke der Partition abgestimmt ist. Einige Speichersubsysteme unterstützen die Rückforderung einer Blockgröße, die höher ist als die LVM-Partitionsgröße, und eine Teilrückfor-

derung von Blöcken ist nicht möglich. In diesem Szenario ist der LVM unter Umständen nicht in der Lage, genügend zusammenhängende freie Partitionen zusammenzustellen, um auch nur eine einzige Rückforderungsanforderung zu generieren. Wenn Sie mehrere LVM-Partitionen löschen, ist es deshalb unter Umständen nicht möglich, den entsprechenden Speicherplatz im Speichersubsystem zurückzufordern. Sie können den Befehl `lvmstat` mit der Option `-r` ausführen, um Informationen zu den vom LVM generierten Anforderungen für die Speicherplatzrückforderung abzurufen.

Zugehörige Informationen:

Befehl `varyoffvg`

Speicherkonzepte für logische Datenträger

Der logische Datenträger (der sich über mehrere physische Datenträger erstrecken kann) setzt sich aus logischen Partitionen zusammen, die physischen Partitionen zugeordnet sind.

Die folgende Abbildung veranschaulicht die Beziehungen zwischen den Basiskonzepten für logischen Speicher.

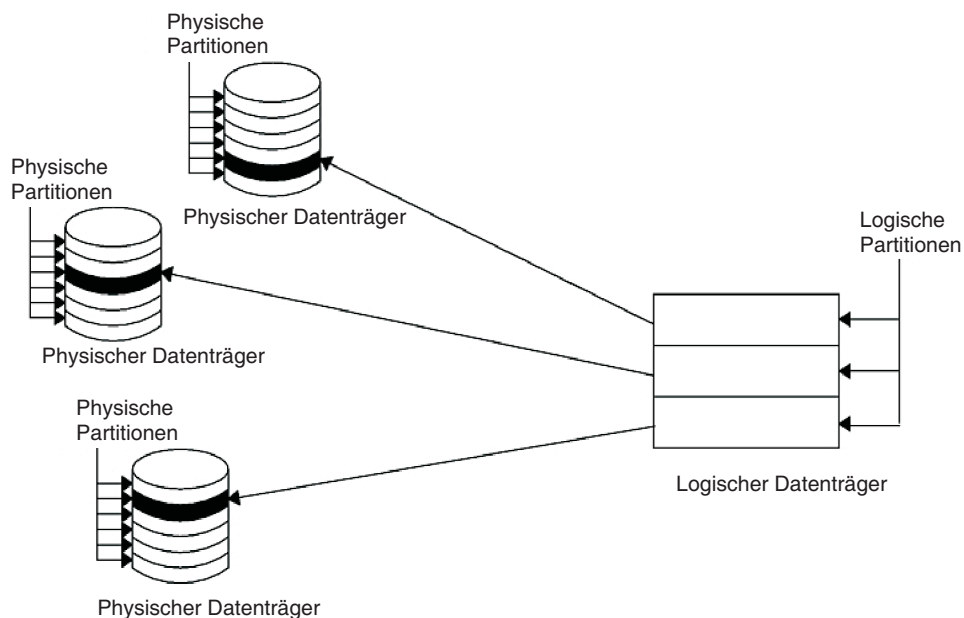


Abbildung 1. Datenträgergruppe. Diese Abbildung zeigt eine Datenträgergruppe, die sich aus drei physischen Datenträgern mit jeweils maximaler Größe zusammensetzt. Der logische Datenträger (der sich über mehrere physische Datenträger erstrecken kann) setzt sich aus logischen Partitionen zusammen, die physischen Partitionen zugeordnet sind.

Physische Datenträger:

Eine Platte muss als physischer Datenträger konfiguriert und in den Status "Verfügbar" (Available) versetzt werden, bevor sie einer Datenträgergruppe zugeordnet werden kann.

Ein physischer Datenträger enthält bestimmte Konfigurations- und Identifikationsdaten. Zu diesen Informationen gehört eine PID (Physical Volume Identifier, ID des physischen Datenträgers), die für das System eindeutig ist.

Der LVM kann den zusätzlichen Speicherplatz, den ein RAID (Redundant Array of Identical Disks) einer LUN (Logical Unit Number, Nummer der logischen Einheit) bietet, nutzen, indem er dem dieser LUN zugeordneten physischen Datenträger physische Partitionen hinzufügt.

Datenträgergruppen:

Eine *Datenträgergruppe* ist eine Sammlung von 1 bis 32 physischen Datenträgern, die in Größe und Typ unterschiedlich sein können.

Eine große Datenträgergruppe kann 1 bis 128 physische Datenträger enthalten. Bei einer skalierbaren Datenträgergruppe sind dies bis zu 1024 physische Datenträger. Ein physischer Datenträger kann nur zu einer Datenträgergruppe pro System gehören. Es kann bis zu 255 aktive Datenträgergruppen geben.

Wenn ein physischer Datenträger einer Datenträgergruppe zugeordnet wird, werden die physischen Blöcke des Speichermediums auf dem Datenträger in physische Partitionen einer Größe gruppiert, die Sie beim Erstellen der Datenträgergruppe angeben.

Wenn Sie das System installieren, wird automatisch eine Datenträgergruppe (die Stammdatenträgergruppe, genannt *rootvg*) erstellt. Diese Datenträgergruppe enthält die Basisgruppe logischer Datenträger, die zum Starten des Systems erforderlich sind, sowie alle weiteren logischen Datenträger, die Sie mit dem Installationsscript angeben. Die Datenträgergruppe *rootvg* enthält auf einem jeweils eigenen logischen Datenträger Paging-Bereich, das Journalprotokoll, Bootdaten und Speicher für Speicherauszüge. Die Attribute der Datenträgergruppe *rootvg* unterscheiden sich von denen benutzerdefinierter Datenträgergruppen. Beispielsweise kann die Datenträgergruppe *rootvg* nicht importiert oder exportiert werden. Wenn Sie einen Befehl oder eine Prozedur für die Datenträgergruppe *rootvg* ausführen, müssen Sie mit den besonderen Merkmalen dieser Datenträgergruppe vertraut sein.

Sie können eine Datenträgergruppe mit dem Befehl **mkvg** erstellen. Mit dem Befehl **extendvg** können Sie einer Datenträgergruppe einen physischen Datenträger hinzufügen. Mit dem Befehl **chvg** können Sie die geänderte Größe eines physischen Datenträgers nutzen, und mit dem Befehl **reducevg** können Sie einen physischen Datenträger aus einer Datenträgergruppe entfernen. Zu den anderen Befehlen, die Sie für Datenträgergruppen verwenden können, gehören **lsvg** (Auflisten), **exportvg** (Entfernen), **importvg** (Installieren), **reorgvg** (Reorganisieren), **syncvg** (Synchronisieren), **varyonvg** (Aktivieren) und **varyoffvg** (Inaktivieren).

Kleine Systeme benötigen möglicherweise nur eine Datenträgergruppe, die alle physischen Datenträger umfasst, die an das System angeschlossen sind. Sie können aus Sicherheitsgründen jedoch separate Datenträgergruppen erstellen, weil jede Datenträgergruppe eigene Sicherheitsberechtigungen haben kann. Außerdem vereinfachen separate Datenträgergruppen die Verwaltung, da andere Gruppen als die, die gewartet wird, aktiv bleiben können. Da die Datenträgergruppe *rootvg* immer online sein muss, enthält sie nur die minimale Anzahl physischer Datenträger, die für den Systembetrieb erforderlich sind.

Mit dem Befehl **migratepv** können Sie Daten von einem physischen Datenträger auf andere physische Datenträger *in derselben Datenträgergruppe* verschieben. Dieser Befehl ermöglicht Ihnen, einen physischen Datenträger freizumachen, damit er aus der Datenträgergruppe entfernt werden kann. Beispielsweise könnten Sie Daten von einem physischen Datenträger entfernen, der ausgetauscht werden muss.

Eine Datenträgergruppe, die mit kleineren Grenzwerten für physische und logische Datenträger erstellt wird, kann in ein Format konvertiert werden, das mehr physische Datenträger und mehr logische Datenträger unterstützt. Diese Operation setzt voraus, dass auf jedem physischen Datenträger in der Datenträgergruppe ausreichend freie Partitionen für die Erweiterung des Deskriptorbereichs der Datenträgergruppe (VGDA, Volume Group Descriptor Area) verfügbar sind. Die Anzahl der erforderlichen freien Partitionen richtet sich nach der Größe des aktuellen VGDA und der Größe der physischen Partitionen. Da der VGDA am Rand der Platte gespeichert ist und zusammenhängenden Speicherplatz benötigt, sind freie Partitionen am Rand der Platte erforderlich. Wenn diese Partitionen einem Benutzer zur Verwendung zugeordnet sind, werden sie in andere freie Partitionen auf derselben Platte verschoben. Die ver-

bleibenden physischen Partitionen werden unnummeriert, um den Verlust der Partitionen für den VGDA widerzuspiegeln. Diese Neunummerierung ändert die Zuordnungen von logischen zu physischen Partitionen auf allen physischen Datenträgern dieser Datenträgergruppe. Wenn Sie die Zuordnungen der logischen Datenträger für eine potenzielle Wiederherstellungsoperation gespeichert haben, müssen Sie die Zuordnungen nach Abschluss der Konvertierungsoperation erneut generieren. Wenn die Sicherung der Datenträgergruppe mit Zuordnungsoption erstellt wird und Sie planen, diese Zuordnungen wiederherzustellen, kann die Wiederherstellungsoperation scheitern, weil die Partitionsnummer (aufgrund der Reduktion) unter Umständen nicht mehr vorhanden ist. Es wird empfohlen, die Sicherung vor der Konvertierung und direkt nach der Konvertierung zu erstellen, wenn die Zuordnungsoption verwendet wird. Da der VGDA erheblich vergrößert wurde, kann jede VGDA-Aktualisierungsoperation (Erstellen eines logischen Datenträgers, Ändern eines logischen Datenträgers, Hinzufügen eines physischen Datenträgers usw.) erheblich länger dauern.

Zugehörige Konzepte:

„Physische Partitionen“

Wenn Sie einer Datenträgergruppe einen physischen Datenträger hinzufügen, wird der physische Datenträger in zusammenhängende Speicherbereichseinheiten gleicher Größe, so genannte *physische Partitionen*, unterteilt. Eine physische Partition ist die kleinste Einheit für die Zuordnung von Speicherbereich und ein zusammenhängender Bereich auf einem physischen Datenträger.

Physische Partitionen:

Wenn Sie einer Datenträgergruppe einen physischen Datenträger hinzufügen, wird der physische Datenträger in zusammenhängende Speicherbereichseinheiten gleicher Größe, so genannte *physische Partitionen*, unterteilt. Eine physische Partition ist die kleinste Einheit für die Zuordnung von Speicherbereich und ein zusammenhängender Bereich auf einem physischen Datenträger.

Physische Datenträger erben die Größe der physischen Partition der Datenträgergruppe, die Sie nur beim Erstellen der Datenträgergruppe (z. B. mit dem Befehl `mkvg -s`) festlegen können. Die folgende Abbildung zeigt die Beziehung zwischen physischen Partitionen auf physischen Datenträgern und Datenträgergruppen.

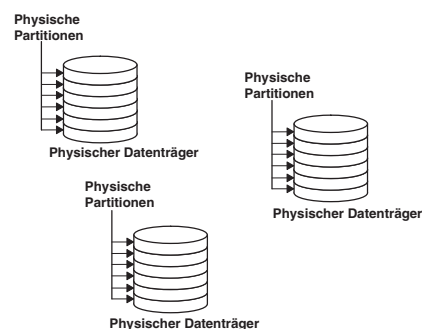


Abbildung 2. Eine Datenträgergruppe mit drei physischen Datenträgern. Diese Abbildung zeigt drei physische Datenträger mit jeweils sechs physischen Partitionen in einer Datenträgergruppe.

Zugehörige Konzepte:

„Datenträgergruppen“ auf Seite 415

Eine *Datenträgergruppe* ist eine Sammlung von 1 bis 32 physischen Datenträgern, die in Größe und Typ unterschiedlich sein können.

Logische Datenträger:

Nachdem Sie eine Datenträgergruppe erstellt haben, können Sie in dieser Datenträgergruppe logische Datenträger erstellen.

Ein *logischer Datenträger* erscheint Benutzern und Anwendungen als einzelner, zusammenhängender, erweiterbarer Plattendatenträger, obwohl er sich auf nicht benachbarten physischen Partitionen oder sogar auf mehreren physischen Datenträgern befinden kann. Weitere logische Datenträger können mit dem Befehl **mklv** erstellt werden. Mit diesem Befehl können Sie den Namen des logischen Datenträgers angeben und seine Merkmale definieren, z. B. Anzahl und Position der logischen Partitionen, die dem Datenträger zugeordnet werden sollen.

Nachdem Sie einen logischen Datenträger erstellt haben, können Sie seinen Namen und seine Merkmale mit dem Befehl **chlv** ändern und die Anzahl der logischen Partitionen, die dem Datenträger zugeordnet sind, mit dem Befehl **extendlv** erhöhen. Die maximale Standardgröße für einen logischen Datenträger bei der Erstellung sind 512 logische Partitionen, sofern kein höherer Wert angegeben wird. Der Befehl **chlv** wird verwendet, um diese Begrenzung zu überschreiben.

Anmerkung: Nach dem Erstellen eines logischen Datenträgers ist der typische Status des logischen Datenträgers (LV STATE), der mit dem Befehl **lslv** angezeigt werden kann, "Geschlossen" (closed). Der Status wechselt in "Geöffnet" (open), wenn beispielsweise ein Dateisystem auf dem logischen Datenträger erstellt und der logische Datenträger angehängt wird.

Logische Datenträger können mit dem Befehl **cplv** kopiert, mit dem Befehl **lslv** aufgelistet und mit dem Befehl **rmlv** entfernt werden. Mit den Befehlen **mklvcopy** und **rmlvcopy** kann die Anzahl der verwalteten Kopien erhöht bzw. verringert werden. Wenn die Datenträgergruppe reorganisiert wird, können logische Datenträger auch verlagert werden.

Sie können auf dem System bis zu 255 logische Datenträger pro Standarddatengruppe erstellen (511 für eine große Datenträgergruppe und 4095 für eine skalierbare Datenträgergruppe), aber die tatsächliche Anzahl, die Sie definieren können, richtet sich nach der physischen Gesamtspeicherkapazität, die für diese Datenträgergruppe definiert ist, und der Größe der logischen Datenträger, die Sie definieren.

Logische Partitionen:

Wenn Sie einen logischen Datenträger erstellen, geben Sie die Anzahl der *logischen Partitionen* für den logischen Datenträger an.

Eine logische Partition setzt sich je nach Anzahl der Instanzen, die Sie von Ihren Daten verwalten möchten, aus einer, zwei oder drei physischen Partitionen zusammen. Wenn Sie eine Instanz angeben, gibt es nur eine Kopie des logischen Datenträgers (Standardeinstellung). In diesem Fall gibt es eine direkte Zuordnung zwischen einer logischen Partition und einer physischen Partition. Jede Instanz, einschließlich der ersten, ist eine *Kopie*. Wo sich die physischen Partitionen befinden (d. h. wie nah beieinander) wird durch die Optionen bestimmt, die Sie beim Erstellen des logischen Datenträgers angeben.

Dateisysteme:

Der logische Datenträger definiert die Zuordnung des Plattenspeicherplatzes bis hin zur Ebene der physischen Partitionen. Differenziertere Stufen der Datenverwaltung werden von Softwarekomponenten der höheren Ebene wie dem Virtual Memory Manager oder dem Dateisystem unterstützt. Deshalb ist der letzte Schritt in der Evolution einer Platte das Erstellen von *Dateisystemen*.

Sie können pro logischem Datenträger ein Dateisystem erstellen. Verwenden Sie zum Erstellen eines Dateisystems den Befehl **crfs**.

Zugehörige Konzepte:

„Dateisysteme“ auf Seite 453

Ein *Dateisystem* ist eine hierarchische Struktur (Dateibaum) von Dateien und Verzeichnissen.

Einschränkungen für die Verwaltung logischen Speichers:

Die folgende Tabelle zeigt die Einschränkungen, die für die Verwaltung logischen Speichers gelten.

Obwohl die maximale Standardanzahl physischer Datenträger pro Datenträgergruppe 32 ist (128 für eine große Datenträgergruppe, 1024 für eine skalierbare Datenträgergruppe), können Sie die Maximalanzahl für benutzerdefinierte Gruppen mit dem Befehl **mkvg** festlegen. Für **rootvg** wird diese Variable jedoch automatisch vom System während der Installation auf den Maximalwert eingestellt.

Einschränkungen für die Verwaltung logischen Speichers

| Kategorie | Grenzwert |
|------------------------|---|
| Datenträgergruppe | <ul style="list-style-type: none"> • 255 Datenträgergruppe für den 32-Bit-Kernel • 4096 Datenträgergruppen für den 64-Bit-Kernel <p>Anmerkung: Die Einheitentabelle des 64-Bit-Kernels begrenzt die Anzahl aktiver Primärzahlen auf 1024. Folglich wird die Zahl der aktiven Datenträgergruppen auf weniger als 1024 Datenträgergruppen eingeschränkt.</p> |
| Physischer Datenträger | (MAXPVS/Datenträgergruppenfaktor) pro Datenträgergruppe. MAXPVS ist 32 für eine Standarddatenträgergruppe, 128 für eine große Datenträgergruppe und 1024 für eine skalierbare Datenträgergruppe. |
| Physische Partition | Normale und große Datenträgergruppen: (1016 x Datenträgergruppenfaktor) pro physischem Datenträger mit jeweils bis zu 1024 MB. Skalierbare Datenträgergruppen: 2097152 Partitionen mit bis zu 128 GB maximal. Es gibt keinen Datenträgergruppenfaktor für skalierbare Datenträgergruppen. |
| Logischer Datenträger | MAXLVS pro Datenträgergruppe. Dies ist 255 für eine Standarddatenträgergruppe, 511 für eine große Datenträgergruppe und 4095 für eine skalierbare Datenträgergruppe. |

Wenn Sie vor der Umsetzung der Einschränkung auf 1016 physische Partitionen pro physischem Datenträger eine Datenträgergruppe erstellt haben, werden veraltete Partitionen (die nicht mehr aktuelle Daten enthalten) in der Datenträgergruppe nur dann ordnungsgemäß verwaltet, wenn Sie die Datenträgergruppe in einen unterstützten Zustand versetzen. Sie können hierfür den Befehl **chvg -t** verwenden. Es wird standardmäßig ein geeigneter Faktor ausgewählt, um die größte Platte in der Datenträgergruppe unterzubringen.

Wenn Sie beispielsweise eine Datenträgergruppe mit einer Platte mit 9 GB und einer Partitionsgröße von 4 MB erstellt haben, hat diese Datenträgergruppe ungefähr 2250 Partitionen. Wenn Sie einen Konvertierungsfaktor von 3 verwenden ($1016 * 3 = 3048$), können alle 2250 Partitionen korrekt verwaltet werden. Die Konvertierung einer Standard- oder großen Datenträgergruppe mit einem höheren Faktor ermöglicht das Hinzufügen einer größeren Platte mit einer Partitionsanzahl bis zu $1016 * \text{Faktor}$. Ein höherer Faktor kann auch angegeben werden, wenn Sie die Datenträgergruppe erstellen, um eine größere Platte mit einer kleinen Partitionsgröße hinzuzufügen.

Diese Operationen verringern die Gesamtanzahl an Platten, die Sie einer Datenträgergruppe hinzufügen können. Die neue maximale Anzahl an Platten, die Sie hinzufügen können, wird mit $\text{MAXPVS}/\text{Faktor}$ berechnet. Bei einer regulären Datenträgergruppe verringert ein Faktor von 2 die maximale Plattenanzahl in der Datenträgergruppe auf 16 ($32/2$). Bei einer großen Datenträgergruppe verringert ein Faktor von 2 die maximale Plattenanzahl in der Datenträgergruppe auf 64 ($128/2$).

Größenbegrenzungen für LVM-Einheiten

Die folgenden Begrenzungen gelten für die LVM-Architektur. Wenn eine LVM-Neupositionierung bei Blockfehler erforderlich ist, darf der physische Datenträger (PV, Physical Volume) nicht größer als 128 GB sein. Informationen zu den Größenbegrenzungen für bestimmte Speichereinheiten finden Sie in der Dokumentation zur jeweiligen Speichereinheit.

Die folgenden Größenbegrenzungen gelten für einen 64-Bit-Kernel:

Ursprüngliche Datenträgergruppe

Begrenzung: 1 GB (PP) * 16256 (PPs/PV, Faktor=16) = 15,9 TB

LV-Begrenzung: 1 GB (PP) * 32512 (PPs/VG) = 31,8 TB

Große Datenträgergruppe

PV-Begrenzung: 1 GB (PP) * 65024 (PPs/PV, Faktor=64) = 63,5 TB

LV-Begrenzung: 1 GB (PP) * 130048 (PPs/VG) = 127 TB

SVG PV- & LV-Begrenzung: 128 GB (PP) * 2048 K (PPs/PV) = 256 PB

Die folgenden Größenbegrenzungen gelten für einen 32-Bit-Kernel:

Alle Datenträgergruppentypen

PV-Begrenzung: < 1 TB

LV-Begrenzung: < 1 TB

Logical Volume Storage konfigurieren

Mit Logical Volume (LVS) können Sie Datenträgergruppen spiegeln, einen logischen Datenträger definieren und eine Platte entfernen, während das System aktiv ist.

Eine Datenträgergruppe spiegeln:

In den folgenden Szenarios wird beschrieben, wie eine normale Datenträgergruppe gespiegelt wird.

Die folgenden Anweisungen veranschaulichen, wie Sie eine Datenträgergruppe mit System Management Interface Tool (SMIT) spiegeln.

(Wählen Sie eine Datenträgergruppe im Container **Datenträger** aus, und wählen Sie anschließend im Menü **Ausgewählt** die Option **Spiegeln** aus.) Erfahrene Administratoren können den Befehl **mirrorvg** verwenden.

1. Melden Sie sich als Benutzer mit Rootberechtigung an, und fügen Sie mit dem folgenden Befehl eine Platte zur Datenträgergruppe hinzu:
smit extendvg
2. Spiegeln Sie die Datenträgergruppe auf der neuen Platte. Geben Sie dazu den folgenden SMIT-Direktaufruf ein:
smit mirrorvg
3. Wählen Sie in der ersten Anzeige eine Datenträgergruppe für die Spiegelung aus.
4. In der zweiten Anzeige können Sie Spiegelungsoptionen definieren oder die Standardoptionen akzeptieren. Zur Unterstützung steht Ihnen eine Onlinehilfe zur Verfügung.

Anmerkung: Nachdem Sie die SMIT-Anzeigen ausgefüllt und auf "OK" oder "Beenden" geklickt haben, kann die Ausführung des zugrunde liegenden Befehls erhebliche Zeit dauern. Die Ausführungsdauer richtet sich nach der Fehlerprüfung, der Größe und Anzahl logischer Datenträger in der Datenträgergruppe und der Zeit, die benötigt wird, um die neu gespiegelten logischen Datenträger zu synchronisieren.

Jetzt sind alle an den logischen Datenträgern vorgenommenen Änderungen gemäß Ihren Angaben in den SMIT-Anzeigen gespiegelt.

Zugehörige Tasks:

„Stammdatenträgergruppe spiegeln“

Im Folgenden wird beschrieben, wie die Stammdatenträgergruppe (rootvg) gespiegelt wird.

Stammdatenträgergruppe spiegeln:

Im Folgenden wird beschrieben, wie die Stammdatenträgergruppe (rootvg) gespiegelt wird.

Anmerkung: Die Spiegelung der Stammdatenträgergruppe setzt weitreichende Erfahrungen in der Systemverwaltung voraus. Wenn die Spiegelung nicht ordnungsgemäß durchgeführt wird, kann Ihr System unter Umständen nicht mehr booten.

Im folgenden Szenario befindet sich die Stammdatenträgergruppe `rootvg` auf `hdisk01`, und die Spiegelung wird auf einer Platte mit dem Namen `hdisk11` erstellt:

1. Vergewissern Sie sich, dass `hdisk11` von AIX als Booteinheit unterstützt wird:

```
bootinfo -B hdisk11
```

Wenn dieser Befehl den Wert 1 zurückgibt, wird die ausgewählte Platte von AIX als Booteinheit anerkannt. Jeder andere Wert bedeutet, dass `hdisk11` kein Kandidat für die Spiegelung von `rootvg` ist.

2. Fügen Sie der Stammdatenträgergruppe `rootvg` mit dem folgenden Befehl die Platte `hdisk11` hinzu:

```
extendvg rootvg hdisk11
```

Es können die folgenden Fehlermeldungen ausgegeben werden:

0516-050 Es ist nicht genügend Platz im Deskriptorbereich der Datenträgergruppe vorhanden.

Fügen Sie einen kleineren physischen Datenträger hinzu oder verwenden Sie eine andere Datenträgergruppe.

oder:

0516-1162 extendvg: Achtung: Die Größe der physischen Partition von 16 erfordert die Erstellung von 1084 Partitionen für `hdisk11`. Die Begrenzung für Datenträgergruppe `rootvg` liegt bei 1016 physischen Partitionen je physischem Datenträger. Über den Befehl `chvg` mit der Option `-t` versuchen, die maximale Anzahl der physischen Partitionen je physischem Datenträger für diese Datenträgergruppe zu ändern.

In diesem Fall haben Sie die folgenden Möglichkeiten:

- Sie können die Stammdatenträgergruppe `rootvg` auf einer leeren Platte spiegeln, die bereits zu `rootvg` gehört.
- Sie können eine kleinere Platte verwenden.
- Sie können die maximale Anzahl an Partitionen, die von `rootvg` unterstützt wird, wie folgt ändern:
 - a. Suchen Sie in der Nachricht die Anzahl der physischen Partitionen, die für die Zielplatte erforderlich sind, und die derzeit von `rootvg` unterstützte maximale Anzahl.
 - b. Verwenden Sie den Befehl **`chvg -t`**, um die derzeit in `rootvg` zulässige maximale Anzahl an Partitionen (im vorherigen Beispiel 1016) so zu vervielfachen, dass sich eine Zahl ergibt, die größer ist als die erforderliche Anzahl physischer Partitionen für die Zielplatte (im vorherigen Beispiel 1084). Beispiel:

```
chvg -t 2 rootvg
```
 - c. Setzen Sie den Befehl **`extendvg`** am Anfang von Schritt 2 erneut ab.

3. Spiegeln Sie die Stammdatenträgergruppe unter Verwendung der Option für exakte Zuordnung, die im folgenden Befehl verwendet wird:

```
mirrorvg -m rootvg hdisk11
```

Dieser Befehl inaktiviert das Quorum, wenn es sich bei der Datenträgergruppe um `rootvg` handelt. Wenn Sie nicht die Option für exakte Zuordnung verwenden, müssen Sie sicherstellen, dass sich die neue Kopie des logischen Bootdatenträgers `hd5` aus zusammenhängenden Partitionen zusammensetzt.

4. Initialisieren Sie alle Bootsätze und -einheiten mit dem folgenden Befehl:

```
bosboot -a
```

5. Initialisieren Sie die Bootliste mit dem folgenden Befehl:

```
bootlist -m normal hdisk01 hdisk11
```

Anmerkung:

- a. Obwohl der Befehl **`bootlist`** `hdisk11` als alternative Bootplatte ausweist, ist nicht gewährleistet, dass das System `hdisk11` als Booteinheit verwendet, wenn `hdisk01` ausfällt. In einem solchen Fall müs-

sen Sie möglicherweise vom Produktdatenträger booten. Wählen Sie **Wartung** aus, und setzen Sie den Befehl **bootlist** ohne Angabe der defekten Platte ab.

- b. Wenn Ihr Hardwaremodell den Befehl **bootlist** nicht unterstützt, können Sie **rootvg** trotzdem spiegeln, müssen aber die alternative Bootplatte aktiv auswählen, wenn die ursprüngliche Platte nicht verfügbar ist.

Zugehörige Tasks:

„Eine Datenträgergruppe spiegeln“ auf Seite 419

In den folgenden Szenarios wird beschrieben, wie eine normale Datenträgergruppe gespiegelt wird.

Einen logischen Datenträger ohne Dateisystem für eine Anwendung definieren:

Ein *logischer Datenträger ohne Dateisystem* ist ein Bereich physischen und logischen Plattenspeicherplatzes, der direkt von einer Anwendung wie einer Datenbank oder einer Partition und nicht vom Betriebssystem oder einem Dateisystem gesteuert wird.

Durch das Umgehen des Dateisystems kann die steuernde Anwendung eine bessere Leistung erzielen. Dies gilt insbesondere für Datenbankanwendungen. Der Grad der Leistungssteigerung hängt von verschiedenen Faktoren wie der Größe einer Datenbank oder dem Treiber der Anwendung ab.

Anmerkung: Sie müssen der Anwendung bei Bedarf die zeichen- oder blockorientierte Sondergeräteeinheit für diesen neuen logischen Datenträger ohne Dateisystem bereitstellen. Die Anwendung stellt bei Öffnungs-, Lese- und Schreiboperationen eine Verbindung zu dieser Gerätedatei her.

Achtung: Jeder logische Datenträger enthält im ersten Block einen so genannten LVCB (Logical-Volume Control Block, Steuerblock für logischen Datenträger). Die Größe des LVCB entspricht der Blockgröße der physischen Datenträger in der Datenträgergruppe. Die Daten beginnen im zweiten Block des physischen Datenträgers. Der LVCB auf einem logischen Datenträger ist nicht geschützt. Wenn eine Anwendung den LVCB überschreibt, schlagen Befehle, die den LVCB gewöhnlich aktualisieren, fehl und generieren eine Nachricht. Der logische Datenträger funktioniert unter Umständen trotzdem noch. Das Überschreiben des LVCB kann auch ein zulässiges Ereignis sein, dennoch wird das Überschreiben des LVCB nicht empfohlen.

In den folgenden Anweisungen werden SMIT und die Befehlszeilenschnittstelle verwendet, um einen logischen Datenträger ohne Dateisystem zu definieren.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Melden Sie sich unter einem Benutzernamen mit Rootberechtigung an, und suchen Sie die freien physischen Partitionen, auf denen Sie den logischen Datenträger ohne Dateisystem erstellen können. Geben Sie hierfür den folgenden SMIT-Direktaufruf ein:

```
smit lspv
```

2. Wählen Sie eine Platte aus.
3. Akzeptieren Sie im zweiten Dialogfenster (Status) den Standardwert, und klicken Sie auf **OK**.
4. Multiplizieren Sie den Wert im Feld **Freie PPs** mit dem Wert im Feld **PP-Größe**, um den für den logischen Datenträger ohne Dateisystem verfügbaren Speicherplatz (in MB) auf der ausgewählten Platte zu berechnen. Falls der freie Speicherplatz nicht ausreichend ist, wählen Sie eine andere Platte aus, auf der Sie genügend freien Speicherbereich finden.
5. Beenden Sie SMIT.
6. Erstellen Sie mit dem Befehl **mklv** den logischen Datenträger ohne Dateisystem. Der folgende Befehl erstellt einen logischen Datenträger ohne Dateisystem mit dem Namen **lvdb2003** in der Datenträgergruppe **db2vg** mit 38 physischen Partitionen von jeweils 4 MB.

```
mklv -y lvdb2003 db2vg 38
```

Mit dem Flag `-y` können Sie einen Namen für den logischen Datenträger angeben, falls Sie keinen vom System generierten Namen verwenden möchten.

Jetzt wird der logische Datenträger ohne Dateisystem erstellt. Wenn Sie den Inhalt Ihrer Datenträgergruppe auflisten, wird ein logischer Datenträger ohne Dateisystem mit dem Standardtyp "jfs" (Journaled File System) angezeigt. Dieser Typeneintrag für einen logischen Datenträger ist einfach ein Kennsatz. Er gibt keinen Aufschluss darüber, ob sich auf dem logischen Datenträger ein Dateisystem befindet.

Ziehen Sie die Anleitungen zu Ihrer Anwendung zu Rate, um festzustellen, wie `/dev/rawLVName` geöffnet und dieser Rohspeicherbereich verwendet wird.

Zugehörige Konzepte:

„Logical Volume Manager konfigurieren“ auf Seite 376

Der Logical Volume Manager (LVM) wird zusammen mit dem Basisbetriebssystem installiert und erfordert keine weitere Konfiguration. Es müssen jedoch Platten konfiguriert und als physische Datenträger definiert werden, damit der LVM sie verwenden kann.


Zugehörige Verweise:

„Verwaltungsbefehle und -direktaufrufe für den LVM“ auf Seite 376

In der folgenden Tabelle sind die einfachsten Tasks gruppiert, die Sie möglicherweise für die Verwaltung der vom LVM gesteuerten Entitäten (physische und logische Datenträger, Datenträgergruppen und Dateisysteme) benötigen.

Zugehörige Informationen:

`mklv` command

 [Logical Volume Manager from A to Z: Introduction and Concepts](#)

Spiegelung der Stammdatenträgergruppe entfernen:

Sie können die Spiegelung der Stammdatenträgergruppe entfernen.

Achtung: Das Entfernen von Spiegelungen der Stammdatenträgergruppe setzt weitreichende Erfahrungen in der Systemverwaltung voraus. Wenn diese Prozedur nicht ordnungsgemäß ausgeführt wird, kann Ihr System möglicherweise nicht mehr booten.

Im folgenden Szenario befindet sich die Stammdatenträgergruppe auf `hdisk01` und die Spiegelung auf `hdisk11`. In diesem Beispiel wird die Spiegelung auf `hdisk11` entfernt. Die Prozedur ist dieselbe, egal über welche Platte Sie zuletzt gebootet haben.

1. Verwenden Sie den folgenden Befehl, um die Spiegelung der Stammdatenträgergruppe auf `hdisk11` zu entfernen:

```
unmirrorvg rootvg hdisk11
```

Der Befehl **unmirrorvg** aktiviert erneut das Quorum für die Stammdatenträgergruppe.

2. Verwenden Sie den folgenden Befehl, um die Platte aus der Stammdatenträgergruppe zu entfernen:

```
reducevg rootvg hdisk11
```

3. Verwenden Sie den folgenden Befehl, um den Bootsatz der verbleibenden Platte zu reinitialisieren:

```
bosboot -a -d /dev/hdisk01
```

4. Verwenden Sie den folgenden Befehl, um die Bootliste zu ändern, um die nicht gespiegelte Platte aus der Liste zu entfernen:

```
bootlist -m normal hdisk01
```

Die Spiegelung der Platte wurde entfernt.

Platten entfernen, während das System verfügbar bleibt:

Die folgende Prozedur beschreibt, wie Sie eine Platte mit der Funktion für das Entfernen von Einheiten im laufenden Betrieb entfernen, die es Ihnen ermöglicht, eine Platte zu entfernen, ohne das System auszu-schalten. Diese Funktion ist nur auf bestimmten Systemen verfügbar.

Die Funktion für das Entfernen von Einheiten im laufenden Betrieb ist in den folgenden Situationen hilf-reich:

- Sie möchten aus Sicherheitsgründen oder zu Wartungszwecken eine Platte mit Daten entfernen, die in einer separaten Datenträgergruppe (nicht rootvg) enthalten ist.
- Sie möchten eine Platte permanent aus einer Datenträgergruppe entfernen.
- Sie möchten einen Plattenfehler beheben.

Eine Platte mit Daten entfernen:

Verwenden Sie diese Prozedur, um eine Platte mit Daten zu entfernen, ohne das System auszuschalten.

Die Platte, die Sie entfernen möchten, muss sich in einer separaten Datenträgergruppe befinden, d. h. nicht in der Datenträgergruppe rootvg. Verwenden Sie diese Prozedur, wenn Sie die Platte in ein anderes System versetzen möchten.

1. Geben Sie Folgendes ein, um die Datenträgergruppe aufzulisten, die der zu entfernenden Platte zugeordnet ist:

```
smit lspv
```

Es erscheint eine Ausgabe wie die folgende:

```
PHYSICAL VOLUME:   hdisk2                VOLUME GROUP:   imagesvg
PV IDENTIFIER:     00083772caa7896e VG IDENTIFIER    0004234500004c00000000e9b5cac262
PV STATE:          active
STALE PARTITIONS:  0                ALLOCATABLE:    yes
PP SIZE:           16 megabyte(s)    LOGICAL VOLUMES: 5
TOTAL PPs:         542 (8672 megabytes) VG DESCRIPTORS: 2
FREE PPs:          19 (304 megabytes)  HOT SPARE:      no
USED PPs:          523 (8368 megabytes)
FREE DISTRIBUTION: 00..00..00..00..19
USED DISTRIBUTION: 109..108..108..108..90
```

Der Name der Datenträgergruppe ist im Feld VOLUME aufgelistet. In diesem Beispiel hat die Daten-trägergruppe den Namen imagesvg.

2. Geben Sie Folgendes ein, um sicherzustellen, dass sich die Platte in einer separaten Datenträgergrup-pe (nicht rootvg) befindet:

```
smit lsvg
```

Wählen Sie anschließend die Datenträgergruppe aus, die Ihrer Platte zugeordnet ist (in diesem Bei-spiel imagesvg). Es erscheint eine Ausgabe wie die folgende:

```
VOLUME GROUP:   imagesvg                VG IDENTIFIER:  0004234500004c00000000e9b5cac262
VG STATE:       active                   PP SIZE:        16 megabyte(s)
VG PERMISSION:  read/write                TOTAL PPs:      542 (8672 megabytes)
MAX LVs:        256                     FREE PPs:       19 (304 megabytes)
LVs:           5                         USED PPs:       523 (8368 megabytes)
OPEN LVs:       4                        QUORUM:         2
TOTAL PVs:     1                         VG DESCRIPTORS: 2
STALE PVs:     0                         STALE PPs:      0
ACTIVE PVs:    1                         AUTO ON:        yes
MAX PPs per PV: 1016                     MAX PVs:        32
LTG size:      128 kilobyte(s)           AUTO SYNC:      no
HOT SPARE:     no
```

In diesem Beispiel zeigt das Feld TOTAL PVs an, dass der Datenträgergruppe imagesvg nur ein physischer Datenträger zugeordnet ist. Da alle Daten in dieser Datenträgergruppe auf hdisk2 enthalten sind, kann hdisk2 mit dieser Prozedur entfernt werden.

3. Geben Sie Folgendes ein, um alle Dateisysteme auf den logischen Datenträgern auf der Platte abzuhängen:
smit umountfs
4. Geben Sie zum Inaktivieren der Datenträgergruppe Folgendes ein:
smit varyoffvg
5. Geben Sie zum Exportieren der Datenträgergruppe Folgendes ein:
smit exportvg
6. Geben Sie zum Entfernen der Platte Folgendes ein:
smit rmvdsk
7. Schauen Sie sich die LED-Anzeige für die Platte an, die Sie entfernen möchten. Vergewissern Sie sich, dass die gelbe LED nicht leuchtet.
8. Entfernen Sie die Platte aus dem Gehäuse. Weitere Informationen zum Entfernen von Einheiten finden Sie im Servicehandbuch zu Ihrem System.

Jetzt ist die Platte physisch und logisch aus dem System entfernt. Wenn Sie die Platte permanent entfernen, ist diese Prozedur damit abgeschlossen.

Zugehörige Tasks:

„Eine Datenträgergruppe importieren oder exportieren“ auf Seite 383

Die folgende Tabelle erläutert, wie Sie durch Import und Export eine benutzerdefinierte Datenträgergruppe von einem System auf ein anderes verschieben. (Die Stammdatenträgergruppe rootvg kann nicht exportiert und importiert werden.)

„Platten hinzufügen, während das System verfügbar bleibt“ auf Seite 378

Die folgende Prozedur beschreibt, wie Sie eine Platte mit der Funktion für das Entfernen von Einheiten im laufenden Betrieb aktivieren und konfigurieren. Diese Funktion ermöglicht Ihnen, Platten hinzuzufügen, ohne das System auszuschalten.

Eine Platte ohne Daten entfernen:

Die folgende Prozedur beschreibt, wie Sie eine Platte entfernen, die keine Daten bzw. keine Daten enthält, die Sie aufbewahren möchten.

Achtung: Die folgende Prozedur löscht alle Daten auf der Platte.

1. Geben Sie Folgendes ein, um alle Dateisysteme auf den logischen Datenträgern auf der Platte abzuhängen:
smit umountfs
2. Geben Sie zum Inaktivieren der Datenträgergruppe Folgendes ein:
smit varyoffvg
3. Geben Sie zum Exportieren der Datenträgergruppe Folgendes ein:
smit exportvg
4. Geben Sie zum Entfernen der Platte Folgendes ein:
smit rmvdsk
5. Schauen Sie sich die LED-Anzeige für die Platte an, die Sie entfernen möchten. Vergewissern Sie sich, dass die gelbe LED nicht leuchtet.
6. Entfernen Sie die Platte aus dem Gehäuse. Weitere Informationen zum Entfernen von Einheiten finden Sie im Servicehandbuch zu Ihrem System.

Jetzt ist die Platte physisch und logisch aus dem System entfernt. Wenn Sie die Platte permanent entfernen, ist diese Prozedur damit abgeschlossen.

Zugehörige Tasks:

„Platten hinzufügen, während das System verfügbar bleibt“ auf Seite 378

Die folgende Prozedur beschreibt, wie Sie eine Platte mit der Funktion für das Entfernen von Einheiten im laufenden Betrieb aktivieren und konfigurieren. Diese Funktion ermöglicht Ihnen, Platten hinzuzufügen, ohne das System auszuschalten.

Einen logischen Datenträger durch Entfernen des Dateisystems entfernen:

Die folgende Prozedur beschreibt, wie Sie ein JFS- oder JFS2-Dateisystem, den zugehörigen logischen Datenträger, die zugehörige Zeilengruppe in der Datei `/etc/filesystems` und optional den Mountpunkt (Verzeichnis), über den das Dateisystem angehängt ist, entfernen.

Achtung: Wenn Sie ein Dateisystem entfernen, löschen Sie damit alle Daten im angegebenen Dateisystem und auf dem logischen Datenträger.

Wenn Sie einen logischen Datenträger entfernen möchten, an dem ein anderer Typ von Dateisystem angehängt ist oder der kein Dateisystem enthält, können Sie nur den logischen Datenträger entfernen.

Verwenden Sie die folgende Prozedur, um ein Journaled File System mit SMIT zu entfernen:

1. Hängen Sie das Dateisystem, das sich auf dem logischen Datenträger befindet, mit einem Befehl wie dem folgenden ab:

```
umount /adam/usr/local
```

Anmerkung: Der Befehl **umount** kann nicht für Einheiten verwendet werden, die sich im Gebrauch befinden. Eine Einheit ist im Gebrauch, wenn aus irgendeinem Grund eine Datei geöffnet ist oder wenn sich das aktuelle Verzeichnis eines Benutzers auf dieser Einheit befindet.

2. Geben Sie den folgenden Direktaufruf ein, um das Dateisystem zu entfernen:

```
smit rmfs
```

- 3.

1. Wählen Sie den Namen des Dateisystems aus, das Sie entfernen möchten.
2. Navigieren Sie zum Feld **Mountpunkt entfernen**, und wählen Sie die gewünschte Einstellung aus. Wenn Sie **ja** auswählen, entfernt der zugrunde liegende Befehl auch den Mountpunkt (Verzeichnis), über den das Dateisystem angehängt ist (sofern dieses Verzeichnis leer ist).
3. Drücken Sie die Eingabetaste, um das Dateisystem zu entfernen. SMIT fordert Sie auf, Ihre Absicht, das Dateisystem zu entfernen, zu bestätigen.
4. Bestätigen Sie, dass Sie das Dateisystem entfernen möchten. SMIT zeigt eine Nachricht an, wenn das Dateisystem erfolgreich entfernt wurde.

Jetzt sind das Dateisystem, seine Daten und der zugehörige logische Datenträger vollständig vom System entfernt.

Zugehörige Tasks:

„Einen logischen Datenträger entfernen“

Verwenden Sie diese Prozedur, um einen logischen Datenträger, auf dem sich ein anderer Dateisystemtyp befindet, oder einen logischen Datenträger, der kein Dateisystem enthält, zu entfernen.

Einen logischen Datenträger entfernen:

Verwenden Sie diese Prozedur, um einen logischen Datenträger, auf dem sich ein anderer Dateisystemtyp befindet, oder einen logischen Datenträger, der kein Dateisystem enthält, zu entfernen.

Achtung: Beim Entfernen eines logischen Datenträgers werden alle Daten in den angegebenen Dateisystemen und auf dem logischen Datenträger gelöscht.

Die folgenden Prozeduren beschreiben, wie ein logischer Datenträger und alle zugehörigen Dateisysteme entfernt werden. Sie können diese Prozedur verwenden, um ein Dateisystem (kein JFS) oder einen logischen Datenträger ohne Dateisystem zu entfernen. Zusätzlich zum Entfernen eines logischen Datenträgers beschreiben die Prozeduren, wie Zeilengruppen für Dateisysteme (kein JFS) in der Datei `/etc/filesystems` entfernt werden.

Verwenden Sie die folgende Prozedur, wenn Sie einen logischen Datenträger mit SMIT entfernen möchten:

1. Wenn der logische Datenträger kein Dateisystem enthält, fahren Sie mit Schritt 4 fort.
2. Hängen Sie mit dem folgenden Befehl alle Dateisysteme ab, die dem logischen Datenträger zugeordnet sind:

```
umount /Name_des_Dateisystems
```

`/Name_des_Dateisystems` steht für den vollständigen Pfadnamen eines Dateisystems.

Anmerkung:

- a. Der Befehl **umount** scheitert, wenn das Dateisystem, das Sie versuchen abzuhängen, derzeit im Gebrauch ist. Der Befehl **umount** wird nur ausgeführt, wenn keine Dateien im Dateisystem geöffnet sind und keine aktuellen Verzeichnisse von Benutzern auf dieser Einheit vorhanden sind.
 - b. Ein anderer Name für den Befehl **umount** ist **umount**. Die Namen sind gegeneinander austauschbar.
3. Geben Sie den folgenden Direktaufruf ein, um Informationen aufzulisten, die Sie über Ihre Dateisysteme benötigen:

```
smit lsfs
```

Im Folgenden sehen Sie einen Teil dieser Liste:

| Name | Nodename | Mount Pt | ... |
|--------------|----------|-----------------|-----|
| /dev/hd3 | -- | /tmp | ... |
| /dev/local1v | -- | /adam/usr/local | ... |

4. Vorausgesetzt, dass Standardnamenskonventionen für den zweiten aufgelisteten Eintrag verwendet wurden, hat das Dateisystem den Namen `/adam/usr/local` und der logische Datenträger den Namen `local1v`. Zur Überprüfung geben Sie den folgenden Direktaufruf ein:

```
smit ls1v2
```

Im Folgenden sehen Sie einen Teil dieser Liste:

```
imagesvg:
LV NAME          TYPE      LPs  PPs  PVs  LV STATE    MOUNT POINT
hd3              jfs       4    4    1    open/syncd  /tmp
local1v          mine      4    4    1    closed/syncd /adam/usr/local
```

5. Zum Entfernen des logischen Datenträgers geben Sie in der Befehlszeile den folgenden Direktaufruf ein:

```
smit rmlv
```

6. Wählen Sie den Namen des logischen Datenträgers aus, den Sie entfernen möchten.
7. Navigieren Sie zum Feld **Mountpunkt entfernen**, und wählen Sie die gewünschte Einstellung aus. Wenn Sie **ja** auswählen, entfernt der zugrunde liegende Befehl auch den Mountpunkt (Verzeichnis), über den das Dateisystem angehängt ist (sofern er vorhanden ist und sofern dieses Verzeichnis leer ist).
8. Drücken Sie die Eingabetaste, um den logischen Datenträger zu entfernen. SMIT fordert Sie auf, Ihre Absicht, den logischen Datenträger zu entfernen, zu bestätigen.
9. Bestätigen Sie, dass Sie den logischen Datenträger entfernen möchten. SMIT zeigt eine Nachricht an, wenn der logische Datenträger erfolgreich entfernt wurde.

10. Wenn auf dem logischen Datenträger ein Dateisystem angehängt ist, das kein JFS-Dateisystem ist, entfernen Sie das Dateisystem und die zugehörige Zeilengruppe in der Datei `/etc/filesystems`, wie im folgenden Beispiel gezeigt:

```
rmfs /adam/usr/local
```

Sie können auch den Namen des Dateisystems verwenden:

```
rmfs /dev/local1v
```

Jetzt ist der logische Datenträger entfernt. Wenn auf dem logischen Datenträger ein Dateisystem enthalten war, das kein JFS-Dateisystem war, wurde auch die Zeilengruppe dieses Dateisystems aus der Datei `/etc/filesystems` entfernt.

Zugehörige Tasks:

„Einen logischen Datenträger durch Entfernen des Dateisystems entfernen“ auf Seite 425

Die folgende Prozedur beschreibt, wie Sie ein JFS- oder JFS2-Dateisystem, den zugehörigen logischen Datenträger, die zugehörige Zeilengruppe in der Datei `/etc/filesystems` und optional den Mountpunkt (Verzeichnis), über den das Dateisystem angehängt ist, entfernen.

Größe einer RAID-Datenträgergruppe ändern:

Auf Systemen, die ein RAID (Redundant Array of Independent Disks) verwenden, können Sie mit den Optionen der Befehle **chvg** und **chpv** eine Platte zur RAID-Gruppe hinzufügen und die Größe des physischen Datenträgers ändern, den der LVM verwendet, ohne den Betrieb oder die Verfügbarkeit des Systems zu unterbrechen.

Anmerkung:

1. Dieses Feature ist nicht verfügbar, wenn die Datenträgergruppe im klassischen oder erweiterten Parallelverarbeitungsmodus aktiviert ist.
2. Die Größe der Datenträgergruppe `rootvg` kann mit der folgenden Prozedur nicht geändert werden.
3. Die Größe einer Datenträgergruppe mit einem aktiven Paging-Bereich kann mit der folgenden Prozedur nicht geändert werden.

Die Größe aller Platten in einer Datenträgergruppe wird automatisch überprüft, wenn die Datenträgergruppe aktiviert wird. Das System generiert eine Informationsnachricht, wenn ein Wachstum festgestellt wird.

Die folgende Prozedur beschreibt, wie Platten in einer RAID-Umgebung vergrößert werden:

1. Geben Sie den folgenden Befehl ein, um eine Platte auf Wachstum zu überprüfen und sie gegebenenfalls zu vergrößern:

```
chvg -g Name_der_Datenträgergruppe
```

Name_der_Datenträgergruppe steht für den Namen Ihrer Datenträgergruppe. Dieser Befehl überprüft alle Platten in der Datenträgergruppe. Wenn ein Wachstum für eine der Platten festgestellt wird, versucht der Befehl, dem physischen Datenträger physische Partitionen hinzuzufügen. Gegebenenfalls bestimmt er den entsprechenden Multiplikator für den Grenzwert von 1016 und konvertiert die Datenträgergruppe in eine große Datenträgergruppe.

2. Geben Sie den folgenden Befehl ein, um die Neupositionierung bei Blockfehlern durch den LVM für die Datenträgergruppe zu inaktivieren:

```
chvg -b ny Name_der_Datenträgergruppe
```

Name_der_Datenträgergruppe steht für den Namen Ihrer Datenträgergruppe.

Datenträgergruppenstrategie

Plattenausfälle sind die am häufigsten auftretenden Hardwarefehler im Speichersystem, gefolgt von Adapter- und Netzteilausfällen. Maßgeblich beteiligt am Schutz vor Plattenausfällen ist die Konfiguration der logischen Datenträger.

Zum Schutz vor Adapter- und Netzteilausfällen sollte für jede Datenträgergruppe eine spezielle Hardwarekonfiguration in Erwägung gezogen werden. Eine solche Konfiguration enthält zwei Adapter mit jeweils mindestens einer Platte, die als Spiegelplatte für den jeweils andern Adapter fungiert, und eine konfigurierte Datenträgergruppe ohne Quorum. Die zusätzlichen Kosten für diese Konfiguration rechtfertigen sich nicht für alle Standorte und Systeme. Sie wird nur empfohlen, wenn eine hohe Verfügbarkeit (bis zur letzten Sekunde) ein Muss ist. Je nach Konfiguration können durch eine hohe Verfügbarkeit Hardwarefehler kompensiert werden, die zwischen der letzten Sicherung und der aktuellen Dateneingabe auftreten. Das versehentliche Löschen von Dateien kann durch eine hohe Verfügbarkeit nicht kompensiert werden.

Zugehörige Konzepte:

„Strategie für logische Datenträger“ auf Seite 430

Die im Folgenden beschriebenen Richtlinien helfen Ihnen, eine Strategie für die Verwendung logischer Datenträger festzulegen, die auf eine Kombination von Verfügbarkeit, Leistung und Kosten, die für Ihren Standort angemessen ist, ausgerichtet ist.

Gründe für das Erstellen separater Datenträgergruppen:

Es gibt mehrere Gründe, physische Datenträger in anderen Datenträgergruppen als der Stammdatenträgergruppe (rootvg) zu organisieren.

- Sicherere und einfachere Wartung.
 - Aktualisierungen des Betriebssystems, Neuinstallationen und Wiederherstellungen nach einem Systemabsturz sind sicherer, weil Sie Benutzerdateisysteme vom Betriebssystem trennen können, so dass die Benutzerdateien durch diese Operationen nicht gefährdet werden.
 - Die Wartung ist einfacher, weil Sie das Betriebssystem aktualisieren oder neu installieren können, ohne Benutzerdaten wiederherstellen zu müssen. Vor einer Aktualisierung können Sie beispielsweise eine benutzerdefinierte Datenträgergruppe aus dem System entfernen, indem Sie die zugehörigen Dateisysteme abhängen. Inaktivieren Sie die Datenträgergruppe mit dem Befehl **varyoffvg**, und exportieren Sie die Gruppe anschließend mit dem Befehl **exportvg**. Nach der Aktualisierung der Systemsoftware können Sie die benutzerdefinierte Datenträgergruppe mit dem Befehl **importvg** erneut hinzufügen und anschließend die zugehörigen Dateisysteme erneut anhängen.
- Für physische Partitionen unterschiedlicher Größe. Alle physischen Datenträger in derselben Datenträgergruppe müssen dieselbe Größe haben. Wenn Sie physische Datenträger unterschiedlicher Größe verwenden möchten, müssen Sie alle Datenträger einer Größe in eine eigene Datenträgergruppe stellen.
- Wenn unterschiedliche Quorummerkmale erforderlich sind. Wenn Sie ein Dateisystem haben, für das Sie eine Datenträgergruppe ohne Quorum erstellen möchten, verwalten Sie für diese Daten eine gesonderte Datenträgergruppe. Alle anderen Dateisysteme sollten in Datenträgergruppen mit Quorum verbleiben.
- Aus Sicherheitsgründen. Sie können eine Datenträgergruppe beispielsweise während der Nacht entfernen.
- Zum Versetzen physischer Datenträger von einem System in ein anderes. Wenn Sie für jedes System an einem Adapter, der für mehrere Systeme zugänglich ist, eine gesonderte Datenträgergruppe erstellen, können Sie die physischen Datenträger zwischen den Systemen verschieben, die an diesen Adapter angeschlossen sind, ohne den normalen Betrieb der Systeme zu beeinträchtigen (siehe Beschreibung der Befehle **varyoffvg**, **exportvg**, **importvg** und **varyonvg**).

Hohe Verfügbarkeit bei einem Plattenausfall:

Zu den wichtigsten Mitteln für den Schutz gegen Plattenausfälle gehören die Konfigurationseinstellungen für logische Datenträger, wie z. B. Spiegelung.

Obwohl die Hinweise zu den Datenträgergruppen zweitrangig sind, haben sie erhebliche wirtschaftliche Auswirkungen, weil sie die Anzahl der physischen Datenträger pro Datenträgergruppe beeinflussen.

- Die Konfiguration mit Quorum sorgt standardmäßig dafür, dass die Datenträgergruppe aktiviert bleibt, solange eine Mehrheit (Quorum) von 51 % der Platten verfügbar ist. In den meisten Fällen benötigen Sie mindestens drei Platten mit gespiegelten Kopien in der Datenträgergruppe, um sich vor einem Plattenausfall zu schützen.
- In einer Konfiguration ohne Quorum bleibt die Datenträgergruppe aktiviert, solange ein Deskriptorbereich für Datenträgergruppen (VGDA, Volume Group Descriptor Area) auf der Platte verfügbar ist. Bei dieser Konfiguration benötigen Sie nur zwei Platten mit gespiegelten Kopien in der Datenträgergruppe, um sich vor einem Plattenausfall zu schützen.

Wenn Sie die Anzahl der Platten für die einzelnen Datenträgergruppen festlegen, müssen Sie auch den Platz für die Spiegelung der Daten einplanen. Beachten Sie, dass Sie für die Spiegelung und das Verschieben von Daten nur Platten aus derselben Datenträgergruppe verwenden können. Wenn die Site große Dateisysteme verwendet, kann es später schwierig werden, Plattenspeicherplatz zu finden, in dem die Daten gespiegelt werden können. Bedenken Sie die Auswirkungen von plattenübergreifenden Einstellungen für Kopien logischer Datenträger und platteninterner Zuordnung für einen logischen Datenträger auf die Verfügbarkeit.

Zugehörige Konzepte:

„Datenträgergruppe aktivieren“ auf Seite 371

Das Aktivieren einer Datenträgergruppe (Vary-On) ist einer der Mechanismen, den der LVM verwendet, um sicherzustellen, dass eine Datenträgergruppe zur Verwendung bereit ist und aktuelle Daten enthält.

„Eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum konvertieren“ auf Seite 375

Sie können eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum ändern, wenn Sie eine fortlaufende Verfügbarkeit der Daten gewährleisten möchten, auch wenn kein Quorum erreicht wird.

„Einstellungen für plattenübergreifende Zuordnung für Kopien logischer Datenträger“ auf Seite 435

Die Zuordnung einer einzelnen Kopie eines logischen Datenträgers auf der Platte ist relativ unkompliziert.

„Richtlinien für platteninterne Zuordnung für jeden logischen Datenträger“ auf Seite 437

Die Optionen für die Richtlinie für platteninterne Zuordnung basieren auf den fünf Bereichen einer Platte, in denen sich physische Partitionen befinden können.

Hohe Verfügbarkeit bei einem Adapter- oder Netzteilerausfall:

Führen Sie eine oder mehrere der im Folgenden beschriebenen Aktionen aus, um sich je nach Anforderungen vor einem Ausfall eines Adapters oder Netzteils zu schützen.

- Verwenden Sie zwei Adapter in demselben oder in unterschiedlichen Gehäusen. Wenn Sie die Adapter in unterschiedlichen Gehäusen installieren, schützt dies vor einem Verlust beider Adapter, falls ein Netzteil in einem Gehäuse ausfällt.
- Verwenden Sie zwei Adapter, und schließen Sie an jeden Adapter mindestens eine Platte an. Diese Maßnahme bietet einen Schutz vor dem Ausfall eines Adapters (oder Netzteils, falls sich die Adapter in unterschiedlichen Gehäusen befinden) und bewahrt trotzdem eine Mehrheit in der Datenträgergruppe, sofern mit *Überkreuzspiegelung* (Kopien für eine logische Partition können nicht auf demselben physischen Datenträger gespeichert werden) zwischen den logischen Datenträgern auf der Platte A (Adapter A) und den logischen Datenträgern auf der Platte B (Adapter B) gearbeitet wird. Das bedeutet, dass Sie die logischen Datenträger auf den an Adapter A angeschlossenen Platten auf die Platten kopieren, die an Adapter B angeschlossen sind, und die logischen Datenträger auf den Platten, die an Adapter B angeschlossen sind, auf die Platten kopieren, die an Adapter A angeschlossen sind.
- Konfigurieren Sie alle Platten an beiden Adaptern in derselben Datenträgergruppe. Auf diese Weise stellen Sie sicher, dass mindestens eine Kopie der logischen Datenträger intakt bleibt, wenn ein Adapter bzw. ein Netzteil (falls sich die Adapter in unterschiedlichen Gehäusen befinden) ausfällt.

- Definieren Sie die Datenträgergruppe als Datenträgergruppe ohne Quorum. In diesem Fall kann die Datenträgergruppe so lange aktiv bleiben, wie ein Deskriptorbereich der Datenträgergruppe (VGDA, Volume Group Descriptor Area) auf irgendeiner Platte in der Datenträgergruppe zugänglich ist.
- Wenn die Datenträgergruppe zwei Platten enthält, implementieren Sie eine Überkreuzspiegelung für die Adapter. Sind an jedem Adapter mehrere Platten verfügbar, implementieren Sie eine Doppelspiegelung. In diese Fall erstellen Sie eine gespiegelte Kopie auf einer Platte, die denselben Adapter verwendet, und eine Kopie auf einer Platte, die einen anderen Adapter verwendet.

Zugehörige Konzepte:

„Eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum konvertieren“ auf Seite 375
 Sie können eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum ändern, wenn Sie eine fortlaufende Verfügbarkeit der Daten gewährleisten möchten, auch wenn kein Quorum erreicht wird.

Strategie für logische Datenträger

Die im Folgenden beschriebenen Richtlinien helfen Ihnen, eine Strategie für die Verwendung logischer Datenträger festzulegen, die auf eine Kombination von Verfügbarkeit, Leistung und Kosten, die für Ihren Standort angemessen ist, ausgerichtet ist.

Verfügbarkeit ist die Fähigkeit, auf Daten selbst dann zugreifen zu können, wenn die zugehörige Platte defekt oder nicht zugänglich ist. Die Daten bleiben möglicherweise über Kopien der Daten zugänglich, die auf separaten Platten und Adaptern während des normalen Systembetriebs erstellt und verwaltet wurden. Techniken wie Spiegelung und Verwendung von Hot-Spare-Platten können zur Gewährleistung der Datenverfügbarkeit beitragen.

Leistung ist die durchschnittliche Geschwindigkeit, mit der auf Daten zugegriffen wird. Richtlinien wie Schreiben und Prüfen (write-verify) und Spiegelung erhöhen die Verfügbarkeit, bedeuten aber zusätzliche Last für das System und wirken sich somit auf die Leistung aus. Durch Spiegelung verdoppelt oder verdreifacht sich die Größe des logischen Datenträgers. Im Allgemeinen wirkt sich eine erhöhte Verfügbarkeit nachteilig auf die Leistung aus. Durch Platten-Striping (einheitenübergreifendes Lesen und Schreiben von Daten) kann die Leistung verbessert werden. Das Platten-Striping ist mit Spiegelung zulässig. Sie können Hot-Spot-Probleme erkennen und beheben, die auftreten, wenn einige der logischen Partitionen auf Ihrer Platte so viel E/As aufweisen, dass die Systemleistung spürbar beeinträchtigt ist.

Durch Steuerung der Zuordnung von Daten auf der Platte und zwischen Platten können Sie das Speichersystem optimieren und die bestmögliche Leistung erzielen. Ausführliche Informationen zum Maximieren der Leistung des Speichersystems finden Sie in der Veröffentlichung *Performance Management*.

Verwenden Sie die folgenden Abschnitte, um die Kompromisse zwischen Leistung, Verfügbarkeit und Kosten zu bewerten. Denken Sie daran, dass eine erhöhte Verfügbarkeit häufig mit Leistungseinbußen einhergeht und sich eine höhere Leistung häufig auf die Verfügbarkeit auswirkt. Durch Spiegelung kann die Leistung verbessert werden. Der LVM wählt für Leseoperationen jedoch die Kopie auf der am wenigsten ausgelasteten Platte aus.

Anmerkung: Spiegelung bietet keinen Schutz vor dem Verlust einzelner Dateien, die versehentlich gelöscht werden oder aufgrund von Softwarefehlern verloren gehen. Diese Dateien können nur von herkömmlichen Band- oder Plattensicherungen wiederhergestellt werden.

Zugehörige Konzepte:

„Datenträgergruppenstrategie“ auf Seite 428
 Plattenausfälle sind die am häufigsten auftretenden Hardwarefehler im Speichersystem, gefolgt von Adapter- und Netzteilausfällen. Maßgeblich beteiligt am Schutz vor Plattenausfällen ist die Konfiguration der logischen Datenträger.

Voraussetzungen für Spiegelung und Striping:

Stellen Sie fest, ob die Daten, die auf dem logischen Datenträger gespeichert sind, wertvoll genug sind, um die Kosten für Verarbeitung und Plattenspeicherplatz für die Spiegelung zu rechtfertigen. Wenn Sie ein großes Dateisystem mit sequenziellem Zugriff haben, das leistungskritisch ist, sollten Sie Platten-Striping in Erwägung ziehen.

Leistung und Spiegelung sind nicht immer unvereinbar. Wenn sich die unterschiedlichen Instanzen (Kopien) der logischen Partitionen auf unterschiedlichen physischen Datenträgern befinden, die vorzugsweise an unterschiedliche Adapter angeschlossen sind, kann der LVM die Leseleistung verbessern, indem er die Kopie auf der am wenigsten genutzten Platte liest. Sofern die Platten nicht an unterschiedliche Adapter angeschlossen sind, sind die Kosten für die Schreibleistung immer dieselben, weil alle Kopien aktualisiert werden müssen. Für eine Leseoperation muss nur eine Kopie gelesen werden.

Der AIX-LVM unterstützt die folgenden RAID-Optionen:

Tabelle 63. LVM-Unterstützung für RAID

| Eintrag | Beschreibung |
|------------------|-------------------------|
| RAID 0 | Striping |
| RAID 1 | Spiegelung |
| RAID 10 oder 0+1 | Spiegelung und Striping |

Obwohl durch Spiegelung eine verbesserte Verfügbarkeit des Speichersystems erreicht werden kann, ist diese Methode kein Ersatz für konventionelle Sicherheitsstrategien mit Bändern.

Sie können die Datenträgergruppe `rootvg` spiegeln. Erstellen Sie in diesem Fall jedoch einen separaten logischen Datenträger für Speicherauszüge. Das Erstellen eines Speicherauszugs auf einem gespiegelten logischen Datenträger kann zu einem inkonsistenten Speicherauszug führen. Erstellen Sie auch einen separaten logischen Datenträger für Speicherauszüge, wenn Sie Ihre logischen Paging-Datenträger spiegeln, weil die Standard Speicherauszugseinheit der primäre logische Paging-Datenträger ist.

Wenn Daten auf einer logischen Partition aktualisiert werden, werden normalerweise alle physischen Partitionen, auf denen die jeweilige logische Partition enthalten ist, automatisch aktualisiert. Physische Partitionen können aufgrund von Systemstörungen oder Nichtverfügbarkeit während einer Aktualisierung jedoch *veraltet* sein (d. h. nicht mehr die aktuellen Daten enthalten). Der LVM kann veraltete Partitionen aktualisieren und in einen konsistenten Zustand versetzen, indem er die aktuellen Daten von einer aktuellen physischen Partition auf die veraltete Partition kopiert. Dieser Prozess wird als *Spiegelsynchronisation* bezeichnet. Die Aktualisierung kann stattfinden, wenn das System erneut gestartet wird, wenn der physische Datenträger wieder online ist oder wenn Sie den Befehl `syncvg` absetzen.

Alle Änderungen, die die Zusammenstellung der physischen Partitionen für einen logischen Bootdatenträger betreffen, erfordern die Ausführung des Befehls `bosboot`. Anders ausgedrückt, Sie müssen nach Aktionen, wie z. B. dem Ändern der Spiegelung eines logischen Bootdatenträgers, den Befehl `bosboot` ausführen.

Planungsrichtlinien für gespiegelte Schreibforderungen auf Platten:

Für Daten, die nur eine physische Kopie haben, übersetzt der LVDD die logische Adresse einer Lese- oder Schreibforderung in eine physische Adresse und ruft den entsprechenden physischen Einheiten-treiber für die Bearbeitung der Anforderung auf. Das Verfahren für Daten mit nur einer Kopie und für nicht gespiegelte Daten beinhaltet die Neupositionierung bei Blockfehlern bei Schreibforderungen und die Rückgabe von Fehlern bei Leseanforderungen an den aufrufenden Prozess.

Wenn Sie gespiegelte logische Datenträger verwenden, können die folgenden Planungsrichtlinien für das Schreiben auf die Platte für einen logischen Datenträger mit mehreren Kopien festgelegt werden:

Planungsrichtlinie "Sequenziell"

Führt nacheinander Schreiboperationen in mehrere Kopien oder Spiegelungen durch. Die physischen Partitionen, die die gespiegelten Kopien einer einzelnen logischen Partition darstellen, werden als primäre, sekundäre und tertiäre Kopie definiert. Bei der sequenziellen Planung werden die physischen Partitionen nacheinander beschrieben. Das System wartet, bis die Schreiboperation für die erste physische Partition abgeschlossen ist, bevor es mit der Schreiboperation für die nächste beginnt. Nachdem alle Schreiboperationen für alle Spiegelungen durchgeführt wurden, ist die Schreiboperation beendet.

Planungsrichtlinie "Parallel"

Startet die Schreiboperation gleichzeitig für alle physischen Partitionen in einer logischen Partition. Wenn die Schreiboperation für die physische Partition, die am längsten dauert, durchgeführt wurde, ist die Schreiboperation abgeschlossen. Die Festlegung der Planungsrichtlinie "Parallel" für gespiegelte logische Datenträger kann die Leistung von E/A-Leseoperationen verbessern, weil das System mit der Verfügbarkeit mehrerer Kopien in der Lage ist, die Leseoperation an die am wenigsten ausgelastete Platte für diesen logischen Datenträger weiterzuleiten.

Planungsrichtlinie "Paralleles Schreiben mit sequenziellem Lesen"

Startet die Schreiboperation gleichzeitig für alle physischen Partitionen in einer logischen Partition. Die primäre Kopie für die Leseoperation wird immer zuerst gelesen. Wenn diese Leseoperation nicht erfolgreich ist, wird die nächste Kopie gelesen. Während der wiederholten Leseoperation mit der nächsten Kopie wird die fehlerhafte primäre Kopie vom LVM durch Hardwareneuordnung korrigiert. Auf diese Weise wird der ungültige Block für künftigen Zugriff korrigiert.

Planungsrichtlinie "Parallel mit Lesen im Round-Robin-Verfahren"

Startet die Schreiboperation gleichzeitig für alle physischen Partitionen in einer logischen Partition. Bei den Leseoperationen wird zwischen den gespiegelten Kopien hin- und hergewechselt.

Richtlinie für fehlerhafte Blöcke

Gibt an, ob die Datenträgergruppe für die Neupositionierung bei Blockfehlern aktiviert ist. Der Standardwert ist *ja*. Wenn der Wert für die Datenträgergruppe auf *ja* gesetzt ist, können fehlerhafte Blöcke neu positioniert werden. Beim Wert *nein* überschreibt die Richtlinie die Einstellungen des logischen Datenträgers. Wenn der Wert geändert wird, verwenden alle logischen Datenträger ihre vorherige Einstellung weiter. Der Wert gibt an, ob eine angeforderte E/A an einen neu positionierten Block weitergeleitet werden muss. Wenn der Wert *ja* lautet, lässt die Datenträgergruppe die Neupositionierung bei Blockfehlern zu. Beim Wert *nein* wird keine Neupositionierung bei Blockfehlern durchgeführt. Der LVM führt Softwareneuordnungen nur durch, wenn die Hardwareneuordnung scheitert. Andernfalls hat die Neupositionierung bei Blockfehlern durch den LVM keine Auswirkung.

Anmerkung: Die Neupositionierung bei Blockfehlern ist inaktiviert, sofern die Einstellungen der Richtlinie für fehlerhafte Blöcke für Datenträgergruppe und logischen Datenträger beide den Wert *ja* haben.

Richtlinie für Spiegelkonsistenz bei logischen Datenträgern:

Wenn die Spiegelkonsistenz (MWC, Mirror Write Consistency) aktiviert ist, werden logische Partitionen ermittelt, deren Konsistenz beeinträchtigt werden kann, wenn das System oder die Datenträgergruppe nicht ordnungsgemäß heruntergefahren wird. Wenn die Datenträgergruppe wieder aktiviert wird, werden diese Informationen verwendet, um die logischen Partitionen wieder konsistent zu machen. Dieser Vorgang wird als *aktive Spiegelkonsistenz* bezeichnet.

Wenn ein logischer Datenträger aktive Spiegelkonsistenz verwendet, werden Anforderungen für diesen logischen Datenträger in der Planungsschicht gehalten, bis die MWC-Cacheblöcke auf den physischen Zieldatenträgern aktualisiert werden können. Nachdem die MWC-Cacheblöcke aktualisiert wurden, fahren die Anforderungen mit dem Schreiben der physischen Daten fort. Diese MWC-Cacheblöcke müssen nur auf die Platten geschrieben werden, auf denen sich die Daten letztendlich befinden, damit die Schreiboperation fortgesetzt werden kann.

Die Verwendung aktiver Spiegelkonsistenz kann sich nachteilig auf die Systemleistung auswirken. Nachteilige Auswirkungen können sich durch den Aufwand ergeben, der durch die Protokollierung und das Journaling einer Schreibanforderung entstehen, in der eine logische Überwachungsgruppe (LTG, Logical Track Group) aktiv ist. Die zulässigen LTG-Größen für eine Datenträgergruppe sind 128 KB, 256 KB, 512 KB, 1024 KB, 2 MB, 4 MB, 8 MB und 16 MB.

Anmerkung: Wenn Sie eine LTG mit mehr als 128 KB verwenden möchten, müssen die Platten in der Datenträgergruppe E/A-Anforderungen dieser Größe über die Strategieroutinen der Platte unterstützen. Die LTG ist ein zusammenhängender Block auf dem logischen Datenträger, ausgerichtet an der Größe der LTG. Dieser Aufwand betrifft nur gespiegelte Schreiboperationen.

Die Datenkonsistenz zwischen Spiegelungen muss nur für den Fall gewährleistet werden, dass das System bzw. die Datenträgergruppe abstürzt, bevor das Schreiben in alle Spiegelungen abgeschlossen ist. Alle logischen Datenträger in einer Datenträgergruppe nutzen dasselbe MWC-Protokoll. Das MWC-Protokoll wird am äußeren Rand einer Platte verwaltet. Positionieren Sie die logischen Datenträger, die aktive Spiegelkonsistenz verwenden, am äußeren Rand der Platte, so dass der logische Datenträger sich auf der Platte in der Nähe des MWC-Protokolls befindet.

Wenn die Spiegelkonsistenz auf passiv eingestellt wird, protokolliert die Datenträgergruppe, dass der logische Datenträger geöffnet wurde. Wenn die Datenträgergruppe nach einem Absturz aktiviert wird, wird automatisch eine Synchronisation auf dem logischen Datenträger erzwungen. Die Konsistenz während der erzwungenen Synchronisation wird durch die Verwendung einer Kopie der Wiederherstellungsrichtlinie für Leseoperationen gewährleistet, die dafür sorgt, dass die gelesenen Blöcke an die anderen Spiegelungen auf dem logischen Datenträger weitergegeben werden. Diese Richtlinie wird nur für große Datenträgergruppen unterstützt.

Wenn die aktive Spiegelkonsistenz inaktiviert wird, können die Spiegelungen eines gespiegelten logischen Datenträgers in einem inkonsistenten Zustand verbleiben, falls ein System oder eine Datenträgergruppe abstürzt. Es gibt keinen automatischen Schutz der Spiegelkonsistenz. Zum Zeitpunkt des Absturzes noch nicht abgeschlossene Schreiboperationen können Spiegelungen mit inkonsistenten Daten hinterlassen, wenn die Datenträgergruppe wieder aktiviert wird. Nach einem Absturz sollte für jeden gespiegelten logischen Datenträger, auf dem die Spiegelkonsistenz inaktiviert ist, eine Synchronisation erzwungen werden, bevor die Daten auf dem logischen Datenträger verwendet werden. Verwenden Sie beispielsweise den folgenden Befehl:

```
syncvg -f -l LTVName
```

Ausgenommen von der erzwungenen Synchronisation sind logische Datenträger, deren Inhalt nur gültig ist, während der logische Datenträger geöffnet ist, z. B. Paging-Bereiche.

Ein gespiegelter logischer Datenträger ist in Bezug auf Schreiboperationen mit einem nicht gespiegelten Datenträger identisch. Wenn der LVM eine Schreibanforderung vollständig beendet, werden die Daten auf alle Laufwerke geschrieben, die der LVM verwaltet. Das Ergebnis der Schreiboperation ist unbekannt, bis der LVM eine Nachricht **iodone** für die Schreiboperation absetzt. Danach ist keine Wiederherstellung nach einem Absturz erforderlich. Alle Blöcke, die (bei **iodone**) nicht vollständig geschrieben wurden, wenn ein System abstürzt, müssen geprüft und erneut geschrieben werden, unabhängig davon, wie die MWC-Einstellung ist oder ob sie gespiegelt sind.

Da sich ein gespiegelter logischer Datenträger nicht von einem nicht gespiegelten Datenträger unterscheidet, gibt es so etwas wie aktuelle Daten nicht. Alle Anwendungen, die Wert auf Datengültigkeit legen, müssen die Gültigkeit der Daten ausstehender oder unvollständiger Schreiboperationen, die vor dem Absturz der Datenträgergruppe bzw. des Systems nicht abgeschlossen wurden, prüfen, egal ob der logische Datenträger gespiegelt wurde oder nicht.

Durch aktive und passive Spiegelkonsistenz kann die Konsistenz der Spiegelungen nur gewährleistet werden, wenn die Datenträgergruppe nach einem Absturz wieder aktiviert wird, indem eine Spiegelung ausgewählt wird und diese Daten an die anderen Spiegelungen weitergegeben werden. Diese MWC-

Richtlinien überwachen nicht die aktuellen Daten. Bei der aktiven Spiegelkonsistenz werden nur die momentan geschriebenen LTGs überwacht. Deshalb ist bei der Spiegelkonsistenz nicht gewährleistet, dass die aktuellen Daten an alle Spiegelungen weitergegeben werden. Die passive Spiegelkonsistenz gewährleistet die Konsistenz von Spiegelungen, indem nach einem Absturz in den Modus "Weitergabe beim Lesen" gewechselt wird. Die dem LVM übergeordnete Anwendung muss die Gültigkeit der Daten nach einem Absturz bestimmen. Wenn die Anwendung nach dem Absturz stets alle ausstehenden Schreibanforderungen erneut absetzt, sind die unter Umständen inkonsistenten Spiegelungen aus der Perspektive des LVM konsistent, sobald diese Schreibanforderungen abgeschlossen sind (solange nach dem Absturz dieselben Blöcke geschrieben werden, die zum Zeitpunkt des Absturzes noch ausstanden).

Anmerkung: Gespiegelte logische Datenträger, die JFS-Protokolle oder -Dateisysteme enthalten, müssen nach einem Absturz durch eine erzwungene Synchronisation synchronisiert werden, bevor sie verwendet werden, indem die aktive oder passive Spiegelkonsistenz aktiviert wird.

Richtlinien für plattenübergreifende Zuordnung:

Die Richtlinie für die plattenübergreifende Zuordnung gibt die Anzahl der Platten an, auf denen die physischen Partitionen eines logischen Datenträgers platziert werden sollen.

Die physischen Partitionen für einen logischen Datenträger können sich auf einer einzelnen Platte befinden oder auf alle Platten in einer Datenträgergruppe verteilt sein. Die folgenden Optionen werden mit den Befehlen **mklv** und **chlv** verwendet, um die Richtlinie für plattenübergreifende Zuordnung zu bestimmen:

- Mit der Option **Range** wird die Anzahl der Platten angegeben, die für eine einzelne physische Kopie des logischen Datenträgers verwendet wird.
- Die Option **Strict** bestimmt, ob die Operation **mklv** erfolgreich ist, wenn zwei oder mehr Kopien auf demselben physischen Datenträger platziert werden müssen.
- Die Option **Super Strict** gibt an, dass die Partitionen, die einer Spiegelkopie zugeordnet sind, nicht auf demselben physischen Datenträger platziert werden können wie die Partitionen einer anderen Spiegelkopie.
- Für einheitübergreifende logische Datenträger kann nur eine Richtlinie für plattenübergreifende Zuordnung mit den Optionen **Range** (maximal) und **Super Strict** verwendet werden.

Zugehörige Konzepte:

„hd6-Paging-Bereich verschieben und verkleinern“ auf Seite 448

Sie können den Standard-Paging-Bereich verkleinern oder verschieben, um die Leistung des Speichersystems durch erzwungenes Paging und Auslagern von Seiten auf andere Platten im System, die weniger ausgelastet sind, zu verbessern. Durch das Verkleinern oder Verschieben des Standard-Paging-Bereichs wird außerdem Plattenspeicherplatz auf `hdisk0` eingespart.

Einstellungen für plattenübergreifende Zuordnung für eine einzelne Kopie des logischen Datenträgers:

Wenn Sie die Einstellung **minimum** für die plattenübergreifende Zuordnung (**Range = minimum**) auswählen, befinden sich die physischen Partitionen, die dem logischen Datenträger zugeordnet sind, auf einer einzigen Platte, um die Verfügbarkeit zu erhöhen. Wenn Sie die Einstellung **Range = maximum** auswählen, befinden sich die physischen Partitionen auf mehreren Platten, um die Leistung zu erhöhen.

Verwenden Sie für nicht gespiegelte logische Datenträger die Einstellung **minimum**, um die höchste Verfügbarkeit zu erzielen (Zugriff auf die Daten bei einem Hardwarefehler). Die Einstellung **minimum** gibt an, dass, sofern möglich, ein physischer Datenträger alle ursprünglichen physischen Partitionen dieses logischen Datenträgers enthält. Wenn das Zuordnungsprogramm zwei oder mehr physische Datenträger verwenden muss, verwendet es bei Gewährleistung der Konsistenz mit anderen Parametern die mögliche minimale Anzahl.

Durch die Verwendung der minimalen Anzahl an physischen Datenträgern können Sie das Risiko eines Datenverlustes im Falle eines Plattenausfalls reduzieren. Jeder zusätzliche physische Datenträger, der für

eine einzelne physische Kopie verwendet wird, erhöht dieses Risiko. Bei einem nicht gespiegelten logischen Datenträger, der auf vier physische Datenträger verteilt ist, ist das Risiko eines Datenverlustes beim Ausfall eines physischen Datenträgers vier Mal höher als bei einem logischen Datenträger auf einem physischen Datenträger.

Die folgende Abbildung veranschaulicht eine Richtlinie für plattenübergreifende Zuordnung mit der Range-Einstellung minimum.

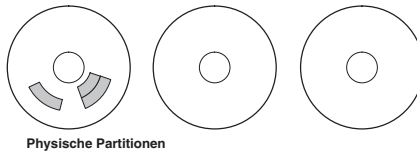


Abbildung 3. Richtlinie für plattenübergreifende Zuordnung mit der Einstellung "minimum"

Diese Abbildung zeigt drei Platten. Eine Platte enthält drei physische Partitionen, die anderen haben keine physischen Partitionen.

Bei der Range-Einstellung maximum werden unter Berücksichtigung anderer Einschränkungen die physischen Partitionen des logischen Datenträgers so gleichmäßig wie möglich auf so viele physische Datenträger wie möglich verteilt. Diese Option ist leistungsorientiert, weil die Verteilung der physischen Partitionen auf mehrere Platten die durchschnittliche Zugriffszeit für den logischen Datenträger verringert. Zur Verbesserung der Verfügbarkeit wird die Einstellung maximum nur für gespiegelte logische Datenträger verwendet.

Die folgende Abbildung veranschaulicht eine Richtlinie für plattenübergreifende Zuordnung mit der Range-Einstellung maximum.

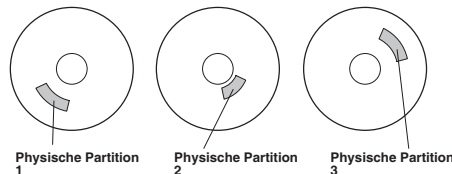


Abbildung 4. Richtlinie für plattenübergreifende Zuordnung mit der Einstellung "maximum"

Diese Abbildung zeigt drei Platten mit jeweils einer physischen Partition.

Diese Definitionen sind auch gültig, wenn ein vorhandener logischer Datenträger erweitert oder kopiert wird. Die Zuordnung neuer physischer Partitionen richtet sich nach der aktuellen Zuordnungsrichtlinie und den Positionen vorhandener verwendeter physischer Partitionen.

Zugehörige Konzepte:

„Einstellungen für plattenübergreifende Zuordnung für Kopien logischer Datenträger“

Die Zuordnung einer einzelnen Kopie eines logischen Datenträgers auf der Platte ist relativ unkompliziert.

Einstellungen für plattenübergreifende Zuordnung für Kopien logischer Datenträger:

Die Zuordnung einer einzelnen Kopie eines logischen Datenträgers auf der Platte ist relativ unkompliziert.

Wenn Sie gespiegelte Kopien erstellen, ist die daraus resultierende Zuordnung etwas komplexer. Die folgenden Abbildungen veranschaulichen die Einstellungen *minimum*, *maximum* und *inter-disk (Range)* für die erste Instanz eines logischen Datenträgers zusammen mit den verfügbaren Einstellungen für die Option *Strict* für die gespiegelten Kopien logischer Datenträger.

Wenn es beispielsweise gespiegelte Kopien des logischen Datenträgers gibt, bewirkt die Einstellung *minimum*, dass die physischen Partitionen mit der ersten Instanz des logischen Datenträgers einem einzigen physischen Datenträger zugeordnet werden, sofern dies möglich ist. Anschließend werden weitere Kopien je nach Einstellung der Option *Strict* demselben oder unterschiedlichen physischen Datenträgern zugeordnet. Anders ausgedrückt, der Algorithmus verwendet innerhalb der Einschränkungen, die durch andere Parameter, wie z. B. die Option *Strict*, auferlegt werden, die minimal mögliche Anzahl an physischen Datenträgern, um alle physischen Partitionen zu speichern.

Die Einstellung *Strict = y* bedeutet, dass jede Kopie der logischen Partition auf einen anderen physischen Datenträger gestellt wird. Die Einstellung *Strict = n* bedeutet, dass die Kopien nicht unbedingt auf unterschiedliche physische Datenträger gestellt werden müssen. Die Option *Super Strict* würde im Vergleich dazu nicht zulassen, dass eine physische Partition einer Spiegelkopie auf dieselbe Platte wie eine physische Partition einer anderen Spiegelkopie desselben logischen Datenträgers gestellt wird.

Anmerkung: Wenn weniger physische Datenträger in der Datenträgergruppe enthalten sind als die ausgewählte Anzahl von Kopien pro logischer Partition, setzen Sie die Option *Strict* auf *n*. Hat *Strict* den Wert *y*, wird eine Fehlermeldung zurückgegeben, wenn Sie versuchen, den logischen Datenträger zu erstellen.

Die folgende Abbildung veranschaulicht eine Richtlinie für plattenübergreifende Zuordnung mit den Einstellungen *minimum* für *Range* und unterschiedlichen Einstellungen für *Strict*:

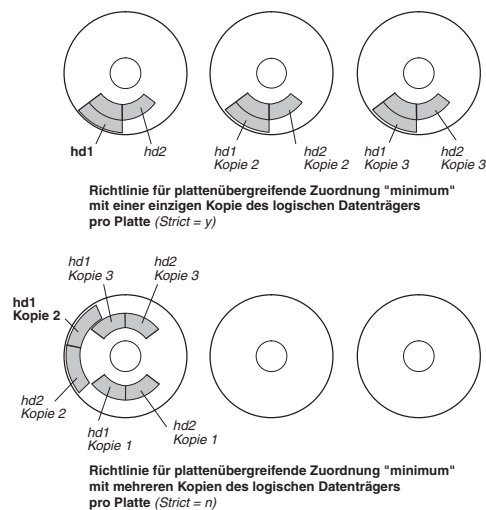


Abbildung 5. Minimum/Strict. Diese Abbildung zeigt, dass jede Kopie der logischen Partition auf einem anderen physischen Datenträger platziert wird, wenn die Option *Strict* den Wert *Yes* hat. Ist *Strict* auf *No* gesetzt, werden alle Kopien der logischen Partitionen auf einen einzigen physischen Datenträger gestellt.

Die folgende Abbildung veranschaulicht eine Richtlinie für plattenübergreifende Zuordnung mit den Einstellungen *maximum* für *Range* und unterschiedlichen Einstellungen für *Strict*:

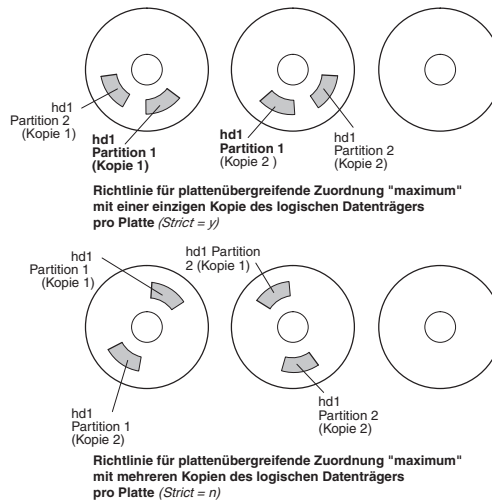


Abbildung 6. Maximum/Strict. Diese Abbildung zeigt, dass sich jede Kopie einer Partition auf einem anderen physischen Datenträger befindet, wenn die Option Strict den Wert Yes hat. Ist Strict auf No gesetzt, werden alle Kopien auf einen einzigen physischen Datenträger gestellt.

Zugehörige Konzepte:

„Hohe Verfügbarkeit bei einem Plattenausfall“ auf Seite 428

Zu den wichtigsten Mitteln für den Schutz gegen Plattenausfälle gehören die Konfigurationseinstellungen für logische Datenträger, wie z. B. Spiegelung.

„Einstellungen für plattenübergreifende Zuordnung für eine einzelne Kopie des logischen Datenträgers“ auf Seite 434

Wenn Sie die Einstellung minimum für die plattenübergreifende Zuordnung (Range = minimum) auswählen, befinden sich die physischen Partitionen, die dem logischen Datenträger zugeordnet sind, auf einer einzigen Platte, um die Verfügbarkeit zu erhöhen. Wenn Sie die Einstellung Range = maximum auswählen, befinden sich die physischen Partitionen auf mehreren Platten, um die Leistung zu erhöhen.

Richtlinien für platteninterne Zuordnung für jeden logischen Datenträger:

Die Optionen für die Richtlinie für platteninterne Zuordnung basieren auf den fünf Bereichen einer Platte, in denen sich physische Partitionen befinden können.

Je näher sich eine bestimmte physische Partition am Zentrum eines physischen Datenträgers befindet, desto geringer ist die durchschnittliche Suchzeit, weil der Suchabstand vom Zentrum zu jedem anderen Teil der Platte durchschnittlich am kürzesten ist.

Das Dateisystemprotokoll ist ein geeigneter Kandidat für das Zentrum eines physischen Datenträgers, weil es so häufig vom Betriebssystem verwendet wird. Der logische Bootdatenträger hingegen wird nur selten verwendet und deshalb am Rand oder in der Mitte des physischen Datenträgers platziert.

Als allgemeine Regel gilt: Je mehr E/As (absolut oder während der Ausführung einer wichtigen Anwendung) die physischen Partitionen des logischen Datenträgers verzeichnen, desto näher am Zentrum der physischen Datenträger sollten sie platziert werden.

Diese Regel hat eine wichtige Ausnahme: Gespiegelte logische Datenträger, bei denen die Einstellung Mirror Write Consistency (MWC) auf On gesetzt ist, befinden sich am äußeren Rand, weil das System die MWC-Daten dorthin schreibt. Wenn die Spiegelung inaktiviert ist, findet MWC keine Anwendung und hat keine Auswirkung auf die Leistung.

Die fünf Regionen, in denen sich physische Partitionen befinden, sind im Folgenden aufgelistet:

1. äußerer Rand
2. innerer Rand
3. äußere Mitte
4. innere Mitte
5. Zentrum

Die Randpartitionen haben die langsamsten durchschnittlichen Suchzeiten, was in der Regel zu längeren Antwortzeiten bei allen Anwendungen führt, die diese Partitionen verwenden. Die Partitionen im Zentrum haben die schnellsten durchschnittlichen Suchzeiten, was in der Regel zu den besten Antwortzeiten bei allen Anwendungen führt, die diese Partitionen verwenden. Im Zentrum eines physischen Datenträgers gibt es jedoch weniger Partitionen als in den äußeren Bereichen.

Zugehörige Konzepte:

„Hohe Verfügbarkeit bei einem Plattenausfall“ auf Seite 428

Zu den wichtigsten Mitteln für den Schutz gegen Plattenausfälle gehören die Konfigurationseinstellungen für logische Datenträger, wie z. B. Spiegelung.

Zuordnungsrichtlinien kombinieren:

Wenn Sie plattenübergreifende und platteninterne Richtlinien auswählen, die nicht kompatibel sind, können Sie unvorhersehbare Ergebnisse erhalten.

Das System ordnet physische Partitionen zu, indem es zulässt, dass eine Richtlinie eine Vorrangstellung gegenüber einer anderen einnimmt. Wenn Sie beispielsweise als platteninterne Richtlinie "Mitte" und als plattenübergreifende Richtlinie "Minimum" auswählen, hat die plattenübergreifende Richtlinie die Vorrangstellung. Das System platziert alle Partitionen für den logischen Datenträger, sofern möglich, auf eine Platte, selbst wenn die Partitionen nicht alle in den mittleren Bereich passen. Sie müssen die Interaktion der Richtlinien unbedingt verstehen, bevor Sie sich für deren Implementierung entscheiden.

Zuordnungsdateien für präzise Zuordnung verwenden:

Wenn die in den Richtlinien für plattenübergreifende und platteninterne Speicherung angegebenen Standardoptionen für Ihre Anforderungen nicht ausreichen, können Sie Zuordnungsdateien erstellen, um die exakte Reihenfolge und Position der physischen Partitionen für einen logischen Datenträger anzugeben.

Sie können für das Erstellen von Zuordnungsdateien SMIT oder den Befehl **mklv -m** verwenden.

Wenn Sie beispielsweise einen logischen Datenträger mit zehn Partitionen mit dem Namen lv06 in der Datenträgergruppe rootvg erstellen möchten und die Partitionen 1 bis 3, 41 bis 45 und 50 bis 60 von hdisk1 zugeordnet werden sollen, können Sie die folgende Prozedur über die Befehlszeile ausführen.

1. Geben Sie Folgendes ein, um zu prüfen, ob die physischen Partitionen, die Sie verwenden möchten, für die Zuordnung verfügbar sind:

```
lspv -p hdisk1
```

2. Erstellen Sie eine Datei, z. B. /tmp/mymap1, die Folgendes enthält:

```
hdisk1:1-3  
hdisk1:41-45  
hdisk1:50-60
```

Der Befehl **mklv** ordnet die physischen Partitionen in der Reihenfolge zu, in der sie in der Zuordnungsdatei angegeben sind. Stellen Sie sicher, dass genügend physische Partitionen in der Zuordnungsdatei angegeben sind, damit der vollständige logische Datenträger, den Sie mit dem Befehl **mklv** angeben, zugeordnet werden kann. (Sie können mehr Partitionen auflisten, als Sie benötigen.)

3. Geben Sie den folgenden Befehl ein:

```
mklv -t jfs -y lv06 -m /tmp/mymap1 rootvg 10
```

Strategien für einheitenübergreifende logische Datenträger entwickeln:

Einheitenübergreifende logische Datenträger werden für große sequenzielle Dateisysteme verwendet, auf die häufig zugegriffen wird und die leistungsabhängig sind. Durch einheitenübergreifendes Lesen und Schreiben von Daten (Striping) kann die Leistung verbessert werden.

Anmerkung: Ein Speicherauszugsbereich oder logischer Bootdatenträger kann nicht einheitenübergreifend verteilt werden. Der logische Bootdatenträger muss aus zusammenhängenden physischen Partitionen bestehen.

Wenn Sie einen einheitenübergreifenden logischen Datenträger mit 12 Partitionen und dem Namen lv07 in VG-Name mit einer Stripe-Größe (die Strip-Größe multipliziert mit der Anzahl der Platten in einem Array entspricht der Stripe-Größe) von 16 KB auf hdisk1, hdisk2 und hdisk3 erstellen möchten, geben Sie Folgendes ein:

```
mk1v -y lv07 -S 16K VG-Name 12 hdisk1 hdisk2 hdisk3
```

Wenn Sie einen einheitenübergreifenden logischen Datenträger mit 12 Partitionen und dem Namen lv08 in VG-Name mit einer Strip-Größe von 8 KB auf drei beliebige Platten in VG-Name erstellen möchten, geben Sie Folgendes ein:

```
mk1v -y lv08 -S 8K -u 3 VG-Name 12
```

Weitere Informationen zur Verbesserung der Leistung mit Platten-Striping finden Sie in der Veröffentlichung *Performance Management*.

Richtlinien für Schreiben-Prüfen (Write-Verify):

Wenn Sie die Option "write-verify" verwenden, werden alle Schreiboperationen durch eine direkt nachfolgende Leseoperation geprüft, um sicherzustellen, dass die Schreiboperation erfolgreich war.

Wenn die Schreiboperation nicht erfolgreich ist, wird eine Fehlernachricht ausgegeben. Diese Richtlinie erhöht die Verfügbarkeit, wirkt sich wegen der zusätzlich benötigten Zeit für die Leseoperation jedoch auf die Leistung aus. Sie können die Verwendung einer write-verify-Richtlinie für einen logischen Datenträger festlegen, wenn Sie den Datenträger mit dem Befehl **mk1v** erstellen oder später mit dem Befehl **chlv** ändern.

Richtlinien für Hot-Spare-Platten:

Sie können Platten als Hot-Spare-Platten für eine Datenträgergruppe mit gespiegelten logischen Datenträgern konfigurieren.

Wenn Sie festlegen, welche Platten als Hot-Spare-Platten verwendet werden sollen, können Sie eine Richtlinie definieren, die verwendet wird, wenn Platten anfangen auszufallen. Sie können auch Synchronisationsmerkmale angeben.

Wenn Sie einer Datenträgergruppe einen physischen Datenträger hinzufügen (um sie als Hot-Spare-Platte zu kennzeichnen), muss die Platte mindestens dieselbe Kapazität wie die kleinste der bereits in der Datenträgergruppe vorhandenen Platten haben. Wenn dieses Feature implementiert ist, werden die Daten auf eine Hot-Spare-Platte migriert, wenn ein physischer Datenträger aufgrund von MWC-Schreibfehlern (Mirror Write Consistency, Spiegelkonsistenz) als fehlend gekennzeichnet wird.

Die Befehle für die Aktivierung der Hot-Spare-Unterstützung, **chvg** und **chpv**, stellen mehrere Optionen für die Implementierung dieses Feature an Ihrem Standort bereit. Schauen Sie sich die folgende Syntax an:

```
chvg -hHot-Spare-Richtlinie -sSynchronisationsrichtlinie Datenträgergruppe
```

Die *Hot-Spare-Richtlinie* bestimmt, welche der folgenden Richtlinien verwendet werden soll, wenn eine Platte ausfällt:

- y** Partitionen werden automatisch von einer fehlerhaften Platte auf eine Hot-Spare-Platte migriert. Aus dem Pool der Hot-Spare-Platten wird die kleinste Platte ausgewählt, die groß genug ist, um die fehlerhafte Platte zu ersetzen.
- Y** Partitionen werden automatisch von einer fehlerhaften Platte migriert, aber es wird möglicherweise der vollständige Pool von Hot-Spare-Platten verwendet.
- n** Es wird keine automatische Migration durchgeführt (Standardeinstellung).
- r** Es werden alle Platten aus dem Pool der Hot-Spare-Platten für diese Datenträgergruppe entfernt.

Das Argument *Synchronisationsrichtlinie* bestimmt, ob veraltete Partitionen automatisch synchronisiert werden:

- y** Es wird automatisch versucht, veraltete Partitionen zu synchronisieren.
- n** Es wird nicht automatisch versucht, veraltete Partitionen zu synchronisieren. (Standardeinstellung)

Das Argument *Datenträgergruppe* gibt den Namen der zugeordneten gespiegelten Datenträgergruppe an.

Hot-Spot-Verwaltung in logischen Datenträgern:

Sie können *Hot-Spot*-Probleme auf logischen Datenträgern identifizieren und diese Probleme beheben, ohne die Systemnutzung zu unterbrechen.

Ein Hot-Spot-Problem tritt auf, wenn einige der logischen Partitionen auf Ihrer Platte so viel Platten-E/As aufweisen, dass die Systemleistung erheblich darunter leidet.

Der erste Schritt bei der Problembekämpfung ist die Identifizierung des Problems. Standardmäßig erfasst das System keine Statistiken zur Verwendung logischer Datenträger. Wenn Sie die Erfassung dieser Statistiken aktivieren, zeigt das System nach der erstmaligen Eingabe des Befehls **lvmstat** die Zählerwerte seit dem letzten Warmstart des Systems an. Wenn Sie den Befehl **lvmstat** danach eingeben, zeigt das System die Differenzwerte zur vorherigen Ausführung des Befehls **lvmstat** an.

Durch die Interpretation der Ausgabe des Befehls **lvmstat** können Sie die logischen Partitionen ermitteln, die den meisten Datenverkehr aufweisen. Wenn Sie mehrere logische Partitionen mit einer hohen Belastung einer physischen Platte haben und diese auf die verfügbaren Platten verteilen möchten, können Sie den Befehl **migratelp** verwenden, um diese logischen Partitionen auf andere physische Platten zu verschieben.

Im folgenden Beispiel ist die Erfassung von Statistiken aktiviert, und der Befehl **lvmstat** wird wiederholt verwendet, um eine Ausgangsversion der Statistiken zu erfassen:

```
# lvmstat -v rootvg -e
# lvmstat -v rootvg -C
# lvmstat -v rootvg
```

Es erscheint eine Ausgabe wie die folgende:

| Logical Volume | iocnt | Kb_read | Kb_wrtn | Kbps |
|----------------|-------|---------|---------|------|
| hd8 | 4 | 0 | 16 | 0.00 |
| paging01 | 0 | 0 | 0 | 0.00 |
| lv01 | 0 | 0 | 0 | 0.00 |
| hd1 | 0 | 0 | 0 | 0.00 |
| hd3 | 0 | 0 | 0 | 0.00 |
| hd9var | 0 | 0 | 0 | 0.00 |
| hd2 | 0 | 0 | 0 | 0.00 |

| | | | | |
|-----|---|---|---|------|
| hd4 | 0 | 0 | 0 | 0.00 |
| hd6 | 0 | 0 | 0 | 0.00 |
| hd5 | 0 | 0 | 0 | 0.00 |

Die vorherige Ausgabe zeigt, dass alle Zähler auf null zurückgesetzt wurden. Im folgenden Beispiel werden die Daten zuerst aus dem Verzeichnis /unix in das Verzeichnis /tmp kopiert. Die Ausgabe des Befehls **lvmstat** spiegelt die Aktivität für die Stammdatenträgergruppe (rootvg) wider:

```
# cp -p /unix /tmp
# lvmstat -v rootvg
```

| Logical Volume | iocnt | Kb_read | Kb_wrtn | Kbps |
|----------------|-------|---------|---------|------|
| hd3 | 296 | 0 | 6916 | 0.04 |
| hd8 | 47 | 0 | 188 | 0.00 |
| hd4 | 29 | 0 | 128 | 0.00 |
| hd2 | 16 | 0 | 72 | 0.00 |
| paging01 | 0 | 0 | 0 | 0.00 |
| lv01 | 0 | 0 | 0 | 0.00 |
| hd1 | 0 | 0 | 0 | 0.00 |
| hd9var | 0 | 0 | 0 | 0.00 |
| hd6 | 0 | 0 | 0 | 0.00 |
| hd5 | 0 | 0 | 0 | 0.00 |

Die Ausgabe zeigt Aktivität auf dem logischen Datenträger **hd3** (angehängt über das Verzeichnis /tmp), auf **hd8** (der logische Datenträger für das JFS-Protokoll), auf **hd4** (Stammdatensystem /), auf **hd2** (das Verzeichnis /usr) und auf **hd9var** (das Verzeichnis /var). Die folgende Ausgabe enthält Details zu **hd3** und **hd2**:

```
# lvmstat -l hd3
```

| Log_part | mirror# | iocnt | Kb_read | Kb_wrtn | Kbps |
|----------|---------|-------|---------|---------|------|
| 1 | 1 | 299 | 0 | 6896 | 0.04 |
| 3 | 1 | 4 | 0 | 52 | 0.00 |
| 2 | 1 | 0 | 0 | 0 | 0.00 |
| 4 | 1 | 0 | 0 | 0 | 0.00 |

```
# lvmstat -l hd2
```

| Log_part | mirror# | iocnt | Kb_read | Kb_wrtn | Kbps |
|----------|---------|-------|---------|---------|------|
| 2 | 1 | 9 | 0 | 52 | 0.00 |
| 3 | 1 | 9 | 0 | 36 | 0.00 |
| 7 | 1 | 9 | 0 | 36 | 0.00 |
| 4 | 1 | 4 | 0 | 16 | 0.00 |
| 9 | 1 | 1 | 0 | 4 | 0.00 |
| 14 | 1 | 1 | 0 | 4 | 0.00 |
| 1 | 1 | 0 | 0 | 0 | 0.00 |

Die Ausgabe für eine Datenträgergruppe enthält eine Übersicht über alle E/A-Aktivitäten auf einem logischen Datenträger. Sie ist in die Anzahl der E/A-Anforderungen (iocnt), die gelesenen und geschriebenen Kilobytes (Kb_read bzw. Kb_wrtn) und die übertragenen Daten in KB pro Sekunde (Kbps) aufgeteilt. Wenn Sie die Informationen für einen logischen Datenträger anfordern, erhalten Sie dieselben Informationen, aber einzeln für jede logische Partition. Wenn Sie gespiegelte logische Datenträger haben, erhalten Sie Statistiken für jeden der Spiegeldatenträger. In der vorherigen Beispielausgabe wurden mehrere Zeilen für logische Partitionen ohne Aktivität weggelassen. Die Ausgabe ist immer in absteigender Reihenfolge der Werte in der Spalte iocnt sortiert.

Der Befehl **migratelp** verwendet als Parameter den Namen des logischen Datenträgers, die Nummer der logischen Partition (die in der Ausgabe des Befehls **lvmstat** angezeigt wird) und eine optionale Nummer für eine bestimmte Spiegelkopie. Wenn keine Informationen angegeben werden, wird die erste Spiegelkopie verwendet. Sie müssen den physischen Zieltatenträger für die Verschiebeoperation angeben. Außerdem können Sie die Nummer einer physischen Zielpartition angeben. Bei erfolgreicher Ausführung erscheint eine Ausgabe wie die folgende:

```
# migratelp hd3/1 hdisk1/109
migratelp: Die Spiegelkopie 1 der logischen Partition 1 des logischen Datenträgers
hd3 wurde auf die physische Partition 109 von hdisk1 migriert.
```

Nachdem Sie die Hot-Spot-Funktion für einen logischen Datenträger oder eine Datenträgergruppe aktiviert haben, können Sie Ihre Berichte und Statistiken definieren, Ihre Statistiken anzeigen, logische Partitionen auswählen, die migriert werden sollen, die physische Zielpartition angeben und die Informationen überprüfen, bevor Sie Ihre Änderungen festschreiben.

Richtlinie für Datenträgergruppen implementieren

Nachdem Sie entschieden haben, welche Richtlinien für die Datenträgergruppen Sie verwenden möchten, analysieren Sie Ihre aktuelle Konfiguration, indem Sie in der Befehlszeile den Befehl `lspv` eingeben.

Die Standardkonfiguration umfasst eine einzelne Datenträgergruppe mit mehreren physischen Datenträgern, die an denselben Adapter angeschlossen sind, und weitere unterstützende Hardware. In einer Standardkonfiguration gilt Folgendes: Je mehr Platten in einer Datenträgergruppe mit Quorum enthalten sind, desto höher ist die Chance, dass das Quorum erhalten bleibt, wenn eine Platte ausfällt. In einer Datenträgergruppe ohne Quorum müssen mindestens zwei Platten in der Datenträgergruppe enthalten sein. Gehen Sie wie folgt vor, um Ihre Änderungen an der Richtlinie für Datenträgergruppen zu implementieren:

1. Verwenden Sie die Ausgabe des Befehls `lspv`, um die zugeordneten und freien physischen Datenträger zu prüfen.
2. Stellen Sie ein Quorum sicher, indem Sie mindestens einen physischen Datenträger hinzufügen.
3. Ändern Sie eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum.
4. Rekonfigurieren Sie die Hardware nur, wenn eine hohe Verfügbarkeit gewährleistet sein muss. Anweisungen finden Sie im Servicehandbuch für Ihr System.

Zugehörige Konzepte:

„Eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum konvertieren“ auf Seite 375

Sie können eine Datenträgergruppe in eine Datenträgergruppe ohne Quorum ändern, wenn Sie eine fortlaufende Verfügbarkeit der Daten gewährleisten möchten, auch wenn kein Quorum erreicht wird.

Zugehörige Tasks:

„Platten hinzufügen, während das System verfügbar bleibt“ auf Seite 378

Die folgende Prozedur beschreibt, wie Sie eine Platte mit der Funktion für das Entfernen von Einheiten im laufenden Betrieb aktivieren und konfigurieren. Diese Funktion ermöglicht Ihnen, Platten hinzuzufügen, ohne das System auszuschalten.

Paging-Bereich und virtueller Speicher

AIX verwendet virtuellen Speicher, um mehr Speicher zu adressieren, als physisch im System verfügbar ist.

Die Verwaltung der Speicherseiten im RAM und auf der Platte wird vom VMM (Virtual Memory Manager) vorgenommen. Die virtuellen Speichersegmente werden in Einheiten, so genannte *Seiten*, aufgeteilt. Ein *Paging-Bereich* ist ein Typ eines logischen Datenträgers mit zugeordnetem Plattenspeicherplatz, in dem Informationen gespeichert werden, die sich im virtuellen Speicher befinden, aber auf die derzeit nicht zugegriffen wird. Dieser logische Datenträger hat ein Typattribut mit dem Wert `Paging` und wird gewöhnlich einfach als *Paging-Bereich* oder *Auslagerungsspeicher* bezeichnet. Wenn die Menge freien Arbeitsspeichers im System gering ist, werden Programme oder Daten, die in der letzten Zeit nicht verwendet wurden, aus dem Hauptspeicher in den *Paging-Bereich* verschoben, um Hauptspeicher für andere Aktivitäten freizugeben.

Konzepte für den Paging-Bereich

Ein *Paging-Bereich* ist ein Typ eines logischen Datenträgers mit zugeordnetem Plattenspeicherplatz, in dem Informationen gespeichert werden, die sich im virtuellen Speicher befinden, aber auf die derzeit nicht zugegriffen wird.

Dieser logische Datenträger hat ein Typattribut mit dem Wert `Paging` und wird gewöhnlich einfach als *Paging-Bereich* oder *Auslagerungsspeicher* bezeichnet. Wenn die Menge freien Arbeitsspeichers im System gering ist, werden Programme oder Daten, die in der letzten Zeit nicht verwendet wurden, aus dem Hauptspeicher in den *Paging-Bereich* verschoben, um Hauptspeicher für andere Aktivitäten freizugeben.

Es gibt einen weiteren Typ von Paging-Bereichen, auf den über eine Einheit zugegriffen werden kann, die einen NFS-Server für die Speicherung von Paging-Bereichen verwendet. Damit ein NFS-Client auf diesen Paging-Bereich zugreifen kann, muss auf dem NFS-Server eine Datei erstellt und auf diesen Client exportiert werden. Die Dateigröße entspricht der Größe des Paging-Bereichs für den Client.

Der erforderliche Paging-Bereich richtet sich nach dem Typ der Aktivitäten, die auf dem System ausgeführt werden. Wenn der Paging-Bereich knapp wird, können Prozesse verloren gehen. Wenn kein Paging-Bereich mehr verfügbar ist, stellt der Kernel jeglichen Betrieb ein (Paniksituation). Definieren Sie zusätzlichen Paging-Bereich, wenn Sie feststellen, dass der Paging-Bereich knapp wird.

Der Paging-Bereich wird definiert, indem ein neuer logischer Datenträger für Paging-Bereiche erstellt oder die Größe vorhandener logischer Datenträger für Paging-Bereiche erhöht wird. Um die Größe eines NFS-Paging-Bereichs zu erhöhen, muss die Datei auf dem Server mit den entsprechenden Aktionen auf dem Server vergrößert werden.

Der Gesamtspeicherplatz, der dem System für Paging zur Verfügung steht, ist die Summe aus den Größen aller aktiven logischen Datenträger für Paging-Bereiche.

Zugehörige Konzepte:

„Fehlerbehebung beim Paging-Bereich“ auf Seite 450

Das am häufigsten auftretende Problem in Bezug auf den Paging-Bereich wird durch einen Mangel an zugeordnetem Speicherplatz verursacht.

Richtlinien für die Zuordnung von Paging-Bereich:

Die Umgebungsvariable *PSALLOC* bestimmt, welcher Algorithmus für die Zuordnung von Paging-Bereich verwendet wird: *deferred* oder *early*.

AIX verwendet zwei Modi für die Zuordnung von Paging-Bereich. Der Standardmodus ist *deferred*. Sie können den Zuordnungsmodus auf *early* umstellen, indem Sie den Wert der Umgebungsvariablen *PSALLOC* ändern. Allerdings müssen vor einer solchen Änderung verschiedene Faktoren berücksichtigt werden. Wenn Sie den Zuordnungsalgorithmus *early* verwenden, kann es im ungünstigsten Fall dazu kommen, dass das System abstürzt, weil der gesamte verfügbare Paging-Bereich aufgebraucht wird.

Zuordnungsmodi deferred und early für Paging-Bereiche im Vergleich:

Das Betriebssystem verwendet die Umgebungsvariable *PSALLOC*, um den Mechanismus festzulegen, der für die Zuordnung von Hauptspeicher und Paging-Bereich verwendet wird.

Wenn die Umgebungsvariable *PSALLOC* nicht gesetzt ist, auf *null* oder auf einen anderen Wert als *early* gesetzt ist, verwendet das System den Standardzuordnungsalgorithmus *deferred*.

Der Zuordnungsalgorithmus *deferred* sorgt mit für eine effiziente Nutzung der Plattenressourcen und unterstützt Anwendungen, die einen schlanken Zuordnungsalgorithmus für die Ressourcenverwaltung bevorzugen. Dieser Algorithmus reserviert keinen Paging-Bereich, wenn eine Speicheranforderung gestellt wird. Vielmehr wird die Zuordnung von Plattenblöcken so lange verzögert, bis eine Auslagerung der angeforderten Seite erforderlich ist. Einige Programme reservieren große Mengen an virtuellem Speicher und nutzen dann nur einen Bruchteil davon. Beispiele für solche Programme sind technische Anwendungen, die schlanke Vektoren oder Matrizen als Datenstrukturen verwenden. Auch für echtzeit- und bedarfsorientierte Kernel wie dem im Betriebssystem ist der Zuordnungsalgorithmus "deferred" effizienter.

Dieser Paging-Bereich wird unter Umständen nie genutzt, insbesondere auf Systemen mit einem sehr großen Echtspeicher, wo Paging nur selten stattfindet. Der Algorithmus "deferred" verzögert die Zuordnung von Paging-Bereich, bis die angeforderte Seite ausgelagert werden muss. In diesem Fall wird also kein Paging-Bereich durch verfrühte Zuordnung verschwendet. Diese verzögerte Zuordnung kann dazu führen, dass der Algorithmus "deferred" versucht, mehr Paging-Bereich zuzuordnen, als dem System an Speicherplatz zur Verfügung steht. Diese Situation wird als Überbelegung von Paging-Bereich bezeichnet.

In einem Überbelegungsszenario, in dem der Paging-Bereich erschöpft ist und versucht wird, einen Plattenblock des Paging-Bereichs für die Auslagerung einer Seite zuzuordnen, tritt ein Fehler auf. Das Betriebssystem versucht, einen Systemausfall zu verhindern, indem es Prozesse, die von der Überbelegung des Paging-Bereichs betroffen sind, beendet. Es sendet das Signal **SIGDANGER**, um die Prozesse darüber zu benachrichtigen, dass der freie Paging-Bereich knapp wird. Wenn die Situation mit dem Paging-Bereich einen noch kritischeren Zustand erreicht, sendet das Betriebssystem an ausgewählte Prozesse, die das Signal **SIGDANGER** nicht erhalten haben, ein Signal **SIGKILL**.

Mit der Umgebungsvariablen *PSALLOC* können Sie den Zuordnungsalgorithmus *early* einstellen, der den Paging-Bereich für den ausführenden Prozess reserviert, wenn der Hauptspeicher angefordert wird. Ist zum Zeitpunkt der Anforderung nicht genügend Paging-Bereich verfügbar, sorgt der Zuordnungsalgorithmus "early" dafür, dass die Hauptspeicheranforderung scheitert.

Wenn die Umgebungsvariable *PSALLOC* auf *early* gesetzt wird, wird jedes Programm, das danach in dieser Umgebung gestartet wird (dies gilt nicht für bereits aktive Prozesse), in der Umgebung mit früher Zuordnung ausgeführt. In einer Umgebung mit früher Zuordnung scheitern Schnittstellen wie die Subroutine **malloc** und die Subroutine **brk**, wenn nicht genügend Paging-Bereich reserviert werden kann, wenn die Anforderung gestellt wird.

Prozesse, die in der Umgebung mit früher Zuordnung ausgeführt werden, erhalten kein Signal **SIGKILL**, wenn der Paging-Bereich knapp wird.

Es gibt verschiedene Möglichkeiten, die Umgebungsvariable *PSALLOC* auf *early* zu setzen. Die Methode richtet sich danach, in welchem Umfang Sie die Änderung vornehmen möchten.

Die folgenden Schnittstellensubroutinen für die Zuordnung von Hauptspeicher sind von einem Wechsel zur Umgebung mit früher Zuordnung betroffen:

- **malloc**
- **free**
- **calloc**
- **realloc**
- **brk**
- **sbrk**
- **shmget**
- **shmctl**

Zugehörige Tasks:

„Umgebungsvariable *PSALLOC* für den Zuordnungsmodus *early* konfigurieren“ auf Seite 446

Das Betriebssystem verwendet die Umgebungsvariable *PSALLOC*, um den Mechanismus festzulegen, der für die Zuordnung von Hauptspeicher und Paging-Bereich verwendet wird.

Zuordnungsmodus early:

Der Zuordnungsmodus "early" gewährleistet, dass so viel Paging-Bereich reserviert wird, wie eine Speicherzuordnungsanforderung anfordert. Deshalb ist die richtige Zuordnung von Paging-Bereich auf einer Systemplatte für die effiziente Ausführung von Operationen entscheidend.

Wenn die Menge verfügbaren Paging-Bereichs unter einen bestimmten Schwellenwert fällt, können keine neuen Prozesse gestartet werden, und derzeit aktive Prozesse sind möglicherweise nicht in der Lage, mehr Hauptspeicher anzufordern. Alle Prozesse, die im Standardzuordnungsmodus "deferred" ausgeführt werden, sind in höchstem Maße anfällig für das Signal **SIGKILL**. Da der Betriebssystemkernel manchmal eine Hauptspeicherzuordnung erfordert, ist es außerdem möglich, dass das System abstürzt, weil der gesamte verfügbare Paging-Bereich aufgebraucht wird.

Bevor Sie den Zuordnungsmodus "early" für das gesamte System verwenden, müssen Sie eine angemessene Menge an Paging-Bereich für das System definieren. Der für den Zuordnungsmodus "early" erforderliche Paging-Bereich ist nahezu immer größer als der Paging-Bereich, der für den Standardzuordnungsmodus "deferred" benötigt wird. Wie viel Paging-Bereich definiert werden muss, richtet sich danach, wie Ihr System verwendet wird und welche Programme Sie ausführen. Ein guter Ausgangspunkt für die Festlegung der richtigen Mischung für Ihr System ist, einen Paging-Bereich zu definieren, der vier Mal so groß ist wie der physische Hauptspeicher.

Bestimmte Anwendungen können extreme Mengen von Paging-Bereich erfordern, wenn sie im Zuordnungsmodus "early" ausgeführt werden. Der AIXwindows-Server erfordert derzeit mehr als 250 MB Paging-Bereich, wenn die Anwendung im Zuordnungsmodus "early" ausgeführt wird. Der für eine Anwendung erforderliche Paging-Bereich ist davon abhängig, wie die Anwendung geschrieben ist und wie sie ausgeführt wird.

Die Ausgabe aller Befehle und Subroutinen, die die Paging-Bereichs- und Prozesshauptspeicherbelegung anzeigen, enthält auch den Paging-Bereich, der im Zuordnungsmodus "early" zugeordnet wird. Der Befehl **lsps** verwendet das Flag **-s**, um den gesamten zugeordneten Paging-Bereich anzuzeigen, einschließlich des im Zuordnungsmodus "early" zugeordneten Paging-Bereichs.

Standardgröße des Paging-Bereichs:

Die Standardgröße des Paging-Bereichs wird während der Systemanpassungsphase der AIX-Installation gemäß den folgenden Standards festgelegt.

- Der Paging-Bereich darf nicht weniger als 16 MB einnehmen. Eine Ausnahme ist die Einheit **hd6**, für die eine Mindestgröße von 64 MB gilt.
- Der Paging-Bereich darf nicht mehr als 20 % des Gesamtplattenspeicherplatzes einnehmen.
- Wenn der Realspeicher kleiner ist als 256 MB, hat der Paging-Bereich die doppelte Größe des Realspeichers.
- Ist der Realspeicher größer-gleich 256 MB, hat der Paging-Bereich eine Größe von 512 MB.

Dateien, Befehle und Optionen für Paging-Bereiche:

Die Datei **/etc/swapspaces** gibt die Paging-Bereiche und die Attribute der Paging-Bereiche an.

Ein Paging-Bereich wird der Datei **/etc/swapspaces** hinzugefügt, wenn er mit dem Befehl **mkps** erstellt wird, und er wird aus der Datei **/etc/swapspaces** entfernt, wenn er mit dem Befehl **rmps** gelöscht wird. Die Attribute des Paging-Bereichs in der Datei werden mit dem Befehl **chps -a** oder mit dem Befehl **chps -c** geändert. Dateien, die ein älteres Format verwenden (das keine Attribute für die Größe der Kontrollsumme und die automatische Auslagerung in den Zeilengruppen enthält), werden weiterhin unterstützt. Wenn der Paging-Bereich zu groß ist, können Sie aus dem Paging-Bereich mit dem Befehl **chps -d** logische Partitionen entfernen, ohne einen Warmstart durchführen zu müssen.

Die folgenden Befehle werden verwendet, um den Paging-Bereich zu verwalten:

| Eintrag | Beschreibung |
|----------------|--|
| chps | Ändert die Attribute eines Paging-Bereichs. |
| lsps | Zeigt die Merkmale eines Paging-Bereichs an. |
| mkps | Fügt einen zusätzlichen Paging-Bereich hinzu. Der Befehl mkps verwendet beim Erstellen eines logischen Datenträgers für den Paging-Bereich den Befehl mklv mit einer bestimmten Gruppe von Optionen. Zum Erstellen von NFS-Paging-Bereichen verwendet der Befehl mkps den Befehl mkdev mit einer anderen Gruppe von Optionen. Für NFS-Paging-Bereiche benötigt der Befehl mkps den Hostnamen des NFS-Servers und den Pfadnamen der Datei, die vom Server exportiert wird. |
| rmps | Entfernt einen inaktiven Paging-Bereich. |
| swapoff | Inaktiviert einen oder mehrere Paging-Bereiche, ohne einen Warmstart durchzuführen. Die Informationen im Paging-Bereich werden in andere aktive Paging-Bereiche verschoben. Der inaktivierte Paging-Bereich kann anschließend mit dem Befehl rmps entfernt werden. |

| Eintrag | Beschreibung |
|---------------|---|
| swapon | Aktiviert einen Paging-Bereich. Der Befehl swapon wird in einem frühen Stadium der Systeminitialisierung verwendet, um die erste Paging-Bereichseinheit zu aktivieren. In einer späteren Phase der Initialisierung, wenn andere Einheiten verfügbar sind, wird der Befehl swapon verwendet, um zusätzliche Paging-Bereiche zu aktivieren, so dass Paging-Aktivitäten auf mehreren Einheiten stattfinden können. |

Die Option **paging type** ist für alle Paging-Bereiche auf logischen Datenträgern erforderlich.

Die folgenden Optionen werden verwendet, um die Paging-Leistung eines logischen Datenträgers zu maximieren:

- Zuordnung von Speicherplatz in der Mitte der Platte, um den Weg des Plattenzugriffsarms zu minimieren.
- Verwendung mehrerer Paging-Bereiche auf unterschiedlichen physischen Datenträgern.

Paging-Bereich konfigurieren

Viele Konfigurationstasks können mit SMIT ausgeführt werden. Paging-Bereich und Hauptspeicherzuordnung wird mit der Umgebungsvariablen **PSALLOC** gesteuert.

Paging-Bereich hinzufügen und aktivieren:

Wenn Sie Paging-Bereich für Ihr System verfügbar machen möchten, müssen Sie den Paging-Bereich hinzufügen und aktivieren.

Das Gesamtvolumen für den Paging-Bereich wird häufig empirisch ermittelt. Häufig wird die Größe des Arbeitsspeichers verdoppelt und diese Zahl dann als Zielgröße für den Paging-Bereich verwendet.

Geben Sie über die Schnittstelle SMIT einen der folgenden Direktaufrufe in der Befehlszeile ein:

- Geben Sie zum Auflisten des aktuellen Paging-Bereichs Folgendes ein: `smit lps`
- Geben Sie zum Hinzufügen von Paging-Bereich Folgendes ein: `smit mkps`
- Geben Sie zum Aktivieren von Paging-Bereich Folgendes ein: `smit swapon`

Zugehörige Tasks:

„Paging-Bereich hd6 innerhalb derselben Datenträgergruppe verschieben“ auf Seite 450

Wenn Sie den Standard-Paging-Bereich von `hdisk0` auf eine andere Platte innerhalb derselben Datenträgergruppe verschieben möchten, müssen Sie das System nicht herunterfahren und erneut booten.

Paging-Leistung verbessern:

Wenn Sie die Paging-Leistung verbessern möchten, verwenden Sie mehrere Paging-Bereiche, und verteilen Sie sie, sofern möglich, auf getrennte physische Datenträger.

Es können aber auch mehrere Paging-Bereiche auf demselben physischen Datenträger verwendet werden. Obwohl Sie mehrere physische Datenträger verwenden können, empfiehlt es sich, nur die Platten in der Stammdatenträgergruppe (`rootvg`) auszuwählen, falls Sie nicht gründlich mit Ihrem System vertraut sind.

Umgebungsvariable PSALLOC für den Zuordnungsmodus early konfigurieren:

Das Betriebssystem verwendet die Umgebungsvariable **PSALLOC**, um den Mechanismus festzulegen, der für die Zuordnung von Hauptspeicher und Paging-Bereich verwendet wird.

Die Standardeinstellung ist `late`. Die folgenden Beispiele zeigen unterschiedliche Möglichkeiten auf, mit denen die Umgebungsvariable **PSALLOC** in `early` geändert werden kann. Die ausgewählte Methode richtet sich danach, in welchem Umfang Sie die Änderung anwenden möchten.

- Geben Sie in einer Shellbefehlszeile den folgenden Befehl ein:
`PSALLOC=early;export PSALLOC`

Dieser Befehl bewirkt, dass alle nachfolgenden Befehle in dieser Shell-Sitzung im Zuordnungsmodus "early" ausgeführt werden.

- Fügen Sie den folgenden Befehl zu einer Shellressourcendatei (.shrc oder .kshrc) hinzu:
PSALLOC=early;export PSALLOC

Dieser Eintrag bewirkt, dass alle Prozesse in Ihrer Anmeldesitzung mit Ausnahme der Anmeldeshell im Zuordnungsmodus "early" ausgeführt werden. Diese Methode schützt die Prozesse auch vor dem Signalmechanismus **SIGKILL**.

- Fügen Sie die Subroutine **putenv** in ein Programm ein, um die Umgebungsvariable *PSALLOC* auf *early* zu setzen. Wenn Sie diese Methode verwenden, wird die Umstellung auf das frühe Zuordnungsverhalten beim nächsten Aufruf der Subroutine **exec** wirksam.

Zugehörige Konzepte:

„Zuordnungsmodi deferred und early für Paging-Bereiche im Vergleich“ auf Seite 443

Das Betriebssystem verwendet die Umgebungsvariable *PSALLOC*, um den Mechanismus festzulegen, der für die Zuordnung von Hauptspeicher und Paging-Bereich verwendet wird.

Einen Paging-Bereich ändern oder entfernen:

Ein Paging-Bereich lässt sich mit SMIT leicht ändern, aber das Entfernen eines Paging-Bereichs ist mit mehr Risiken behaftet.

Wenn Sie die Merkmale eines Paging-Bereichs ändern möchten, können Sie dazu den folgenden SMIT-Direktaufruf eingeben: `smit chps`.

Die Prozedur zum Entfernen eines Paging-Bereichs ist mit mehr Risiken behaftet, insbesondere wenn der Paging-Bereich, den Sie entfernen möchten, ein Standard-Paging-Bereich ist, z. B. `hd6`. Für das Entfernen der Standard-Paging-Bereiche ist eine spezielle Prozedur erforderlich, weil diese Paging-Bereiche während des Bootvorgangs von Shell-Scripts aktiviert werden, die das System konfigurieren. Wenn Sie einen der Standard-Paging-Bereiche entfernen möchten, müssen Sie diese Scripts ändern und ein neues Boot-Image erstellen.

Achtung: Werden die Standard-Paging-Bereiche nicht ordnungsgemäß entfernt, wird das System möglicherweise nicht erneut gestartet. Die folgende Prozedur ist nur für erfahrene Systemmanager bestimmt.

Verwenden Sie zum Entfernen eines vorhandenen Paging-Bereichs die folgende Prozedur:

1. Melden Sie sich als Benutzer mit Rootberechtigung an, und geben Sie in der Befehlszeile den folgenden SMIT-Direktaufruf ein, um den Paging-Bereich zu inaktivieren:
`smit swapoff`
2. Wenn der Paging-Bereich, den Sie entfernen möchten, die Standardspeicherauszugseinheit ist, müssen Sie einen anderen Paging-Bereich oder logischen Datenträger als Standardspeicherauszugseinheit definieren, bevor Sie den Paging-Bereich entfernen. Geben Sie den folgenden Befehl ein, um die Standardspeicherauszugseinheit zu ändern:
`sysdumpdev -P -p /dev/neue_Speicherauszugseinheit`
3. Geben Sie den folgenden Direktaufruf ein, um den Paging-Bereich zu entfernen:
`smit rmps`

Zugehörige Konzepte:

„Fehlerbehebung beim Paging-Bereich“ auf Seite 450

Das am häufigsten auftretende Problem in Bezug auf den Paging-Bereich wird durch einen Mangel an zugeordnetem Speicherplatz verursacht.

Programmierschnittstelle für Steuerung des Zuordnungsmodus für Paging-Bereich verwenden:

Die Programmierschnittstelle, die den Zuordnungsmodus für Paging-Bereich steuert, verwendet die Umgebungsvariable *PSALLOC*.

Gehen Sie wie folgt vor, um sicherzustellen, dass eine Anwendung immer im gewünschten Modus (mit oder ohne frühe Paging-Bereichszuordnung) ausgeführt wird:

1. Verwenden Sie die Subroutine **getenv**, um den aktuellen Status der Umgebungsvariablen *PSALLOC* zu untersuchen.
2. Wenn der Wert der Umgebungsvariablen *PSALLOC* nicht dem von der Anwendung geforderten Wert entspricht, können Sie den Wert der Umgebungsvariablen mit der Subroutine **setenv** ändern. Da nur die Subroutine **execve** den Status der Umgebungsvariablen *PSALLOC* untersucht, rufen Sie die Subroutine **execve** mit denselben Parametern und derselben Umgebung auf, die von der Anwendung empfangen wurden. Wenn die Anwendung den Status der Umgebungsvariablen *PSALLOC* erneut untersucht und den richtigen Wert findet, wird die Ausführung der Anwendung normal fortgesetzt.
3. Wenn die Subroutine **getenv** feststellt, dass der aktuelle Status der Umgebungsvariablen *PSALLOC* korrekt ist, ist keine Änderung erforderlich. Die Ausführung der Anwendung wird normal fortgesetzt.

hd6-Paging-Bereich verschieben und verkleinern:

Sie können den Standard-Paging-Bereich verkleinern oder verschieben, um die Leistung des Speichersystems durch erzwungenes Paging und Auslagern von Seiten auf andere Platten im System, die weniger ausgelastet sind, zu verbessern. Durch das Verkleinern oder Verschieben des Standard-Paging-Bereichs wird außerdem Plattenspeicherplatz auf *hdisk0* eingespart.

Egal ob Sie den Paging-Bereich verschieben oder verkleinern, die Begründung ist immer dieselbe: Paging-Bereichsaktivitäten sollen auf Platten mit einer geringeren Auslastung verlagert werden. Bei der Standardinstallation wird ein logischer Paging-Datenträger (*hd6*) auf Laufwerk *hdisk0* erstellt, der die vollständigen Dateisysteme / (Stammdateisystem) und /usr oder einen Teil davon enthält. Diese Dateisysteme sind gewöhnlich sehr belastet. Wenn Sie die Richtlinie *minimum* für plattenübergreifende Zuordnung auswählen, die bewirkt, dass sich das gesamte Stammdateisystem (/) und ein großer Teil von /usr auf *hdisk0* befinden, kann durch das Verschieben des Paging-Bereichs auf eine weniger ausgelastete Platte die Leistung erheblich verbessert werden. Selbst wenn die Richtlinie *maximum* für plattenübergreifende Zuordnung implementiert ist und beide Dateisysteme (/ und /usr) auf mehrere physische Datenträger verteilt sind, enthält Ihre *hdisk2* (mit angenommenen drei Platten) wahrscheinlich weniger logische Partitionen, die zu den am stärksten ausgelasteten Dateisystemen gehören.

Die folgenden Prozeduren beschreiben, wie Sie den Paging-Bereich *hd6* verkleinern und wie Sie den Paging-Bereich *hd6* innerhalb derselben Datenträgergruppe verschieben.

Zugehörige Konzepte:

„Richtlinien für plattenübergreifende Zuordnung“ auf Seite 434

Die Richtlinie für die plattenübergreifende Zuordnung gibt die Anzahl der Platten an, auf denen die physischen Partitionen eines logischen Datenträgers platziert werden sollen.

„Fehlerbehebung beim Paging-Bereich“ auf Seite 450

Das am häufigsten auftretende Problem in Bezug auf den Paging-Bereich wird durch einen Mangel an zugeordnetem Speicherplatz verursacht.

Paging Bereich hd6 verkleinern:

In der folgenden Prozedur wird der Befehl **chps** verwendet, um vorhandene Paging-Bereiche, einschließlich des primären Paging-Bereichs und der primären und sekundären Speicherausgabeeinheiten zu verkleinern.

Der Befehl **chps** ruft das Script **shrinkps** auf, das den Paging-Bereich sicher verkleinert, ohne das System in einen Zustand zu versetzen, in dem es nicht mehr gebootet werden kann. Das Script führt im Einzelnen die folgenden Schritte aus:

1. Es erstellt einen temporären Paging-Bereich auf demselben Datenträger.
2. Es verschiebt Informationen in diesen temporären Bereich.
3. Es erstellt einen neuen, kleineren Paging-Bereich auf demselben Datenträger.
4. Es entfernt den alten Paging-Bereich.

Damit der Befehl **chps** erfolgreich ausgeführt wird, muss ausreichend freier Plattenspeicherplatz (Speicherplatz, der keinem logischen Datenträger zugeordnet ist) vorhanden sein, damit ein temporärer Paging-Bereich erstellt werden kann. Die Größe des temporären Paging-Bereichs entspricht der Menge an Speicherplatz, die erforderlich ist, um alle ausgelagerten Seiten im alten Paging-Bereich zu speichern. Die Mindestgröße für einen primären Paging-Bereich sind 32 MB. Die Mindestgröße für jeden anderen Paging-Bereich sind 16 MB.

Anmerkung: Wenn während der folgenden Prozedur ein E/A-Fehler auftritt, muss das System möglicherweise sofort heruntergefahren und erneut gebootet werden.

1. Überprüfen Sie mit dem folgenden Befehl Ihren logischen Datenträger und die Verteilung des Dateisystems auf die physischen Datenträger:

```
lspv -l hdiskX
```

hdiskX steht für den Namen Ihres physischen Datenträgers.

2. Geben Sie in der Befehlszeile den folgenden Befehl ein, um den Paging-Bereich zu verkleinern:
smit **chps**

Anmerkung: Der primäre Paging-Bereich ist fest im Bootsatz codiert. Deshalb wird der primäre Paging-Bereich bei einem Neustart des Systems immer aktiviert. Der Befehl **chps** kann den primären Paging-Bereich nicht inaktivieren.

Die Aufrechterhaltung einer betriebsfähigen Konfiguration hat oberste Priorität. Systemprüfungen können zu einer sofortigen Verweigerung einer Verkleinerung des Paging-Bereichs führen. Fehler, die während der Erstellung des Paging-Bereichs auftreten, bewirken, dass die Prozedur beendet und das System auf die ursprünglichen Einstellungen zurückgesetzt wird. Andere Probleme können Situationen hervorrufen, die einen Eingriff des Systemadministrators oder unter Umständen einen sofortigen Warmstart erfordern. Einige Fehler können das Entfernen des temporären Paging-Bereichs verhindern. Eine solche Situation erfordert in der Regel einen Eingriff des Administrators, der jedoch nicht sofort erfolgen muss.

Achtung: Wenn der Befehl **swapoff** im Script **shrinkps** einen E/A-Fehler in den System- oder Benutzersicherungsseiten feststellt, wird ein sofortiges Herunterfahren des Systems empfohlen, um einen möglichen Systemabsturz zu verhindern. Beim Warmstart wird der temporäre Paging-Bereich aktiviert, und es kann versucht werden, die Anwendungen, in denen die E/A-Fehler aufgetreten sind, zu stoppen und anschließend erneut zu starten. Wenn der Versuch erfolgreich ist und der Befehl **swapoff** die Inaktivierung abschließen kann, kann die Verkleinerungsprozedur manuell mit den Befehlen **mkps**, **swapoff** und **rmpps** durchgeführt werden, um einen Paging-Bereich der erforderlichen Größe zu erstellen und den temporären Paging-Bereich zu entfernen.

Versuchen Sie nicht, einen inaktivierten Paging-Bereich, für den E/A-Fehler festgestellt wurden, vor dem Systemwiederanlauf (mit dem Befehl **rmpps**) zu entfernen oder (mit dem Befehl **chps**) zu reaktivieren. Es besteht das Risiko, dass der Plattenspeicherplatz wiederverwendet wird, was zu zusätzlichen Problemen führen könnte.

Paging-Bereich hd6 innerhalb derselben Datenträgergruppe verschieben:

Wenn Sie den Standard-Paging-Bereich von hdisk0 auf eine andere Platte innerhalb derselben Datenträgergruppe verschieben möchten, müssen Sie das System nicht herunterfahren und erneut booten.

Geben Sie den folgenden Befehl ein, um den Standard-Paging-Bereich (hd6) von hdisk0 auf hdisk2 zu verschieben:

```
migratepv -l hd6 hdisk0 hdisk2
```

Achtung: Das Verschieben eines Paging-Bereichs mit dem Namen hd6 von der Stammdatenträgergruppe rootvg in eine andere Datenträgergruppe wird nicht empfohlen, weil der Name an mehreren Stellen fest codiert ist, z. B. in der zweiten Phase des Bootprozesses und in dem Prozess, der beim Booten über einen austauschbaren Datenträger auf die Stammdatenträgergruppe zugreift. Während der zweiten Phase des Bootprozesses sind nur die Paging-Bereiche in rootvg aktiv, und wenn in rootvg kein Paging-Bereich vorhanden ist, könnte dies schwerwiegende Auswirkungen auf die Leistung beim Booten des Systems haben. Wenn Sie den größten Teil des Paging-Bereichs in anderen Datenträgergruppen haben möchten, empfiehlt es sich, hd6 so klein wie möglich zu machen (Größe, die dem physischen Speicher entspricht) und anschließend größere Paging-Bereiche in anderen Datenträgergruppen zu erstellen.

Zugehörige Konzepte:

„Paging-Bereich hinzufügen und aktivieren“ auf Seite 446

Wenn Sie Paging-Bereich für Ihr System verfügbar machen möchten, müssen Sie den Paging-Bereich hinzufügen und aktivieren.

Fehlerbehebung beim Paging-Bereich

Das am häufigsten auftretende Problem in Bezug auf den Paging-Bereich wird durch einen Mangel an zugeordnetem Speicherplatz verursacht.

Das Gesamtvolumen für den Paging-Bereich wird häufig empirisch ermittelt. Häufig wird die Größe des Arbeitsspeichers verdoppelt und diese Zahl dann als Zielgröße für den Paging-Bereich verwendet. Wenn der Paging-Bereich knapp wird, können Prozesse verloren gehen. Wenn kein Paging-Bereich mehr verfügbar ist, stellt der Kernel jeglichen Betrieb ein (Paniksituation). Die folgenden Informationen zu Signalen und Fehlern können Ihnen helfen, Probleme mit dem Paging-Bereich zu überwachen, zu beheben und sogar zu verhindern.

Das Betriebssystem überwacht die Anzahl freier Blöcke im Paging-Bereich und erkennt, wenn der Paging-Bereich knapp wird. Wenn die Anzahl der freien Blöcke im Paging-Bereich unter einen Schwellenwert fällt, der als *Warnstufe für den Paging-Bereich* bezeichnet wird, informiert das System alle Prozesse (mit Ausnahme von **kprocs**) über diese Situation, indem es das Signal **SIGDANGER** sendet. Setzt sich diese Situation fort und fällt der freie Paging-Bereich unter einen zweiten Schwellenwert, die so genannte *Beendigungsstufe für den Paging-Bereich*, sendet das System das Signal **SIGKILL** an die Prozesse, die die Hauptnutzer des Paging-Bereichs sind und keine Signalaroutine für das Signal **SIGDANGER** besitzen. (Standardmäßig wird das Signal **SIGDANGER** ignoriert.) Das System sendet daraufhin das Signal **SIGKILL** so oft, bis die Anzahl der freien Blöcke im Paging-Bereich die Beendigungsstufe für den Paging-Bereich wieder überschreitet.

Anmerkung: Wenn der Parameter **low_ps_handling** auf 2 gesetzt ist (siehe Abschnitt zum Befehl **vmo**) und kein Prozess gefunden wird, der (ohne eine Routine für **SIGDANGER**) beendet werden kann, sendet das System das Signal **SIGKILL** an die jüngsten Prozesse, die eine Signalaroutine für das Signal **SIGDANGER** besitzen.

Prozesse, die Hauptspeicher dynamisch zuordnen, können sicherstellen, dass ausreichend Paging-Bereich vorhanden ist, indem sie die Stufen für den Paging-Bereich mit der Subroutine **psdanger** oder mit speziellen Zuordnungsroutinen überwachen. Mit der Subroutine **disclaim** können Sie verhindern, dass Prozesse beendet werden, wenn die Beendigungsstufe für den Paging-Bereich erreicht ist. Hierfür definieren Sie

eine Signalaroutine für das Signal **SIGDANGER** und geben Hauptspeicher und Ressourcen für den Paging-Bereich frei, die Daten- und Stack-Bereichen sowie gemeinsam genutzten Speichersegmenten zugeordnet sind.

Wenn Fehlermeldungen wie die folgenden ausgegeben werden, vergrößern Sie den Paging-Bereich:

```
INIT: Paging space is low!
```

ODER

```
You are close to running out of paging space.  
You may want to save your documents because  
this program (and possibly the operating system)  
could terminate without future warning when the  
paging space fills up.
```

Zugehörige Konzepte:

„hd6-Paging-Bereich verschieben und verkleinern“ auf Seite 448

Sie können den Standard-Paging-Bereich verkleinern oder verschieben, um die Leistung des Speichersystems durch erzwungenes Paging und Auslagern von Seiten auf andere Platten im System, die weniger ausgelastet sind, zu verbessern. Durch das Verkleinern oder Verschieben des Standard-Paging-Bereichs wird außerdem Plattenspeicherplatz auf `hdisk0` eingespart.

„Konzepte für den Paging-Bereich“ auf Seite 442

Ein *Paging-Bereich* ist ein Typ eines logischen Datenträgers mit zugeordnetem Plattenspeicherplatz, in dem Informationen gespeichert werden, die sich im virtuellen Speicher befinden, aber auf die derzeit nicht zugegriffen wird.

Zugehörige Tasks:

„Einen Paging-Bereich ändern oder entfernen“ auf Seite 447

Ein Paging-Bereich lässt sich mit SMIT leicht ändern, aber das Entfernen eines Paging-Bereichs ist mit mehr Risiken behaftet.

Virtual Memory Manager

Virtual Memory Manager (VMM) verwaltet die Speicheranforderungen des Systems und dessen Anwendungen.

Virtuelle Speichersegmente sind in Einheiten, so genannte *Seiten*, untergliedert. Jede Seite befindet sich entweder im realen physischen Speicher (RAM) oder ist so lange auf der Festplatte gespeichert, bis sie benötigt wird. AIX verwendet virtuellen Speicher, um mehr Speicher zu adressieren, als physisch im System verfügbar ist. Die Verwaltung der Speicherseiten im RAM und auf der Platte wird vom VMM vorgenommen.

Realspeicherverwaltung im Virtual Memory Manager:

In AIX sind die Segmente des virtuellen Speichers in 4096-Byte-Einheiten, so genannten Seiten, partitioniert. Der Realspeicher ist in 4096-Byte-Seitenrahmen eingeteilt.

Der VMM (Virtual Memory Manager) hat zwei Hauptfunktionen:

- Er verwaltet die Zuordnung von Seitenrahmen.
- Er löst Referenzen auf Seiten des virtuellen Speichers auf, die derzeit nicht im Arbeitsspeicher enthalten (im Paging-Bereich gespeichert) oder noch nicht vorhanden sind.

Für die Ausführung dieser Funktionen verwaltet der VMM eine *Liste der freien Seitenrahmen*. Außerdem verwendet der VMM einen Seitenersetzungsalgorithmus, um festzustellen, welche Seitenrahmen der virtuellen Speicherseiten, die sich derzeit im Arbeitsspeicher befinden, der Liste der freien Seiten wieder zugeordnet werden. Der Seitenersetzungsalgorithmus berücksichtigt die Existenz von persistenten Segmenten und Ausführungsspeichersegmenten (Working Memory), wiederholtes Paging und VMM-Schwellenwerte.

Liste der freien Seiten des Virtual Memory Manager:

Der Virtual Memory Manager (VMM) verwaltet eine Liste mit den freien (nicht zugeordneten) Seitenrahmen, die er für die Bearbeitung von Fehlseitenbedingungen verwendet.

AIX versucht stets, den gesamten Arbeitsspeicher zu nutzen. Ausgenommen hiervon ist nur ein kleiner Teil des Arbeitsspeichers, den der VMM in der Liste der freien Seiten verwaltet. Für die Verwaltung dieses kleinen Teils nicht zugeordneter Seiten verwendet der VMM *Auslagerungen* und *Speicherseitenübernahmen*, um Platz freizugeben und diese Seitenrahmen der Liste der freien Seiten erneut zuzuordnen. Die Seiten des virtuellen Speichers, deren Seitenrahmen erneut zugeordnet werden sollen, werden mit dem VMM-Algorithmus für Seitenersetzung ausgewählt.

Persistente Speichersegmente und Ausführungsspeichersegmente im Virtual Memory Manager:

AIX unterscheidet zwischen unterschiedlichen Typen von Speichersegmenten. Zum Verständnis des VMM ist es wichtig, den Unterschied zwischen Ausführungsspeichersegmenten (Working Memory) und persistenten Segmenten zu kennen.

Ein *persistentes Segment* hat eine permanente Speicherposition auf der Platte. Dateien, die Daten oder ausführbare Programme enthalten, werden persistenten Segmenten zugeordnet. Wenn eine JFS- oder JFS2-Datei geöffnet und aufgerufen wird, werden die Dateidaten in den Arbeitsspeicher kopiert. VMM-Parameter steuern, wann physische Speicherrahmen, die persistenten Seiten zugeordnet sind, überschrieben und zum Speichern anderer Daten verwendet werden sollen.

Ausführungsspeichersegmente sind temporäre Segmente und existieren nur so lange, wie sie von einem Prozess verwendet werden. Ausführungsspeichersegmente haben keine permanente Speicherposition auf der Platte. Prozessorstapel und Datenbereiche werden Ausführungsspeichersegmenten und Textsegmenten von gemeinsam genutzten Bibliotheken zugeordnet. Seiten von Ausführungsspeichersegmenten müssen Speicherpositionen auf der Platte belegen, wenn sie nicht im Realspeicher gehalten werden können. Der Paging-Bereich auf der Platte wird für diesen Zweck verwendet. Bei der Beendigung eines Programms werden alle Ausführungsspeicherseiten des Programms sofort in die Liste der freien Seiten zurückgestellt.

Ausführungsspeichersegmente und Paging-Bereich in Virtual Memory Manager:

Ausführungsspeicherseiten im RAM, die geändert und ausgelagert werden können, wird ein entsprechender Bereich im Paging-Bereich zugeordnet.

Der zugeordnete Paging-Bereich wird nur verwendet, wenn die Seite ausgelagert werden muss. Eine zugeordnete Seite im Paging-Bereich kann jedoch nicht von einer anderen Seite verwendet werden. Sie bleibt so lange für eine bestimmte Seite reserviert, wie diese Seite im virtuellen Speicher existiert. Da persistente Seiten an dieselbe Plattenposition ausgelagert werden, von der sie stammen, muss für persistente Seiten im RAM kein Paging-Bereich zugeordnet werden.

Der VMM hat zwei Modi für die Zuordnung von Paging-Bereich: *early* und *late*. Die Zuordnungsrichtlinie "early" reserviert Paging-Bereich, wenn eine Speicheranforderung für eine Arbeitsseite gestellt wird. Die Zuordnungsrichtlinie "late" ordnet Paging-Bereich nur zu, wenn die Arbeitsseite wirklich aus dem Hauptspeicher ausgelagert wird, was den Bedarf an Paging-Bereich auf dem System erheblich reduziert.

Steuerungsfunktion für das Laden von Speicher im Virtual Memory Manager:

Wenn ein Prozess auf eine virtuelle Speicherseite verweist, die sich auf der Platte befindet, weil sie ausgelagert oder noch nie gelesen wurde, muss die referenzierte Seite eingelagert werden. Dies kann dazu führen, dass eine oder mehrere Seiten ausgelagert werden müssen, wenn die Anzahl verfügbarer (freier) Seitenrahmen gering ist. Der VMM versucht, unter Verwendung eines Algorithmus für Seitenersetzung Seitenrahmen zu übernehmen (Page Steal), die in letzter Zeit nicht referenziert wurden und wahrscheinlich auch in naher Zukunft nicht referenziert werden.

Bei einer erfolgreichen Seitenersetzung werden die Speicherseiten aller derzeit aktiven Prozesse im Arbeitsspeicher (RAM) gehalten, während die Speicherseiten inaktiver Prozesse ausgelagert werden. Wenn der Arbeitsspeicher jedoch überlastet ist, wird es schwierig, Seiten für die Auslagerung auszuwählen, weil sie wahrscheinlich in Kürze von derzeit aktiven Prozessen referenziert werden. Das Ergebnis ist, dass Seiten, die wahrscheinlich bald referenziert werden, ausgelagert und anschließend, wenn die Referenzierung tatsächlich erfolgt, wieder eingelagert werden. Wenn der Arbeitsspeicher überlastet ist, kann es zu ständigen Ein- und Auslagerungen oder einer *Überlastung* kommen. Wenn ein System überlastet ist, verbringt es die meiste Zeit damit, Seiten ein- und auszulagern, anstatt sinnvolle Anweisungen auszuführen, und keiner der aktiven Prozesse macht einen erheblichen Fortschritt. Der VMM besitzt einen Algorithmus für die Steuerung der Speicherlast, der erkennt, wenn das System überlastet ist, und in einem solchen Fall versucht, das Problem zu beheben.

Dateisysteme

Ein *Dateisystem* ist eine hierarchische Struktur (Dateibaum) von Dateien und Verzeichnissen.

Diese Struktur gleicht einem umgekehrten Baum mit dem Stamm oben und den Ästen (oder Verzweigungen) unten. In diesem Dateibaum werden Verzeichnisse verwendet, um Daten und Programme zu Gruppen zusammenzufassen. Auf diese Weise können mehrere Verzeichnisse und Dateien gleichzeitig verwaltet werden.

Ein Dateisystem befindet sich auf einem logischen Datenträger. Jede Datei und jedes Verzeichnis gehört zu einem Dateisystem auf einem logischen Datenträger. Aufgrund der Struktur des Dateisystems werden einige Tasks effizienter für ein Dateisystem als für ein Verzeichnis innerhalb des Dateisystems ausgeführt. Beispielsweise können Sie ein vollständiges Dateisystem sichern, verschieben und schützen. Sie können ein zeitpunktgesteuertes Image eines JFS-Dateisystems oder eines JFS2-Dateisystems, eine so genannte *Momentaufnahme*, erstellen.

Anmerkung: Die maximale Anzahl logischer Partitionen pro logischem Datenträger ist 32.512. Weitere Informationen zu den Merkmalen logischer Datenträger für Dateisysteme finden Sie in der Beschreibung des Befehls **chlv** in der Veröffentlichung "Commands Reference".

Der Befehl **mkfs** (make file system, Dateisystem erstellen) und System Management Interface Tool (Befehl **smit**) erstellen ein Dateisystem auf einem logischen Datenträger.

Um auf ein Dateisystem zugreifen zu können, muss das Dateisystem an einen Verzeichnismountpunkt angehängt werden. Wenn mehrere Dateisysteme angehängt sind, wird eine Verzeichnisstruktur erstellt, die das Bild eines einzigen Dateisystems vermittelt. Es handelt sich um eine hierarchische Struktur mit einem Stammverzeichnis. Diese Struktur enthält die Basisdateisysteme und alle Dateisysteme, die Sie erstellen. Sie können auf lokale und ferne Dateisysteme mit dem Befehl **mount** zugreifen. Dieser Befehl macht das Dateisystem für Lese- und Schreibzugriffe auf dem lokalen System zugänglich. Für das An- und Abhängen eines Dateisystems wird gewöhnlich die Zugehörigkeit zur Gruppe **system** vorausgesetzt. Dateisysteme können automatisch angehängt werden, wenn sie in der Datei `/etc/filesystems` definiert sind. Sie können ein lokales oder fernes Dateisystem mit dem Befehl **umount** abhängen, sofern kein Benutzer oder Prozess auf das Dateisystem zugreift. Weitere Informationen zum Anhängen eines Dateisystems finden Sie im Abschnitt „Anhängen (Mount)“ auf Seite 482.

Der von AIX verwendete Basisdateisystemtyp ist *Journalled File System (JFS)*. In diesem Dateisystem werden Aufzeichnungstechniken wie bei Datenbanken verwendet, um die strukturelle Konsistenz des Dateisystems zu bewahren. Dadurch werden Beschädigungen am Dateisystem bei abnormalen Systemstopps verhindert.

Das Betriebssystem AIX unterstützt mehrere Dateisystemtypen, einschließlich Journalled File System (JFS) und Enhanced Journalled File System (JFS2). Weitere Informationen zu Dateisystemtypen und den Merkmalen jedes Typs finden Sie im Abschnitt „Dateisystemtypen“ auf Seite 488.

Einige der wichtigsten Systemverwaltungstasks beziehen sich auf Dateisysteme, wie z. B.:

- Speicherplatz für Dateisysteme auf logischen Datenträgern reservieren
- Dateisysteme erstellen
- Systembenutzern Speicherplatz im Dateisystem zur Verfügung stellen
- Speicherplatzauslastung des Dateisystems überwachen
- Dateisysteme zum Schutz vor Datenverlusten bei Systemstörungen sichern
- Konsistenz der Dateisysteme erhalten

Diese Tasks müssen vom Systemadministrator ausgeführt werden.

Zugehörige Konzepte:

„Dateisysteme“ auf Seite 417

Der logische Datenträger definiert die Zuordnung des Plattenspeicherplatzes bis hin zur Ebene der physischen Partitionen. Differenziertere Stufen der Datenverwaltung werden von Softwarekomponenten der höheren Ebene wie dem Virtual Memory Manager oder dem Dateisystem unterstützt. Deshalb ist der letzte Schritt in der Evolution einer Platte das Erstellen von *Dateisystemen*.

Zugehörige Tasks:

„Momentaufnahme eines JFS2 erstellen und sichern“ auf Seite 38

Sie können eine Momentaufnahme (oder Snapshot) eines angehängten JFS2 erstellen, die ein konsistentes Image des Dateisystems auf Blockebene zu einem bestimmten Zeitpunkt liefert.

„Externe Momentaufnahme eines JFS2 erstellen und sichern“ auf Seite 39

Sie können eine Momentaufnahme (oder Snapshot) eines angehängten JFS2 erstellen, die ein konsistentes Image des Dateisystems auf Blockebene zu einem bestimmten Zeitpunkt liefert.

„Interne Momentaufnahme eines JFS2 erstellen und sichern“ auf Seite 40

Sie können eine Momentaufnahme (oder Snapshot) eines angehängten JFS2 erstellen, die ein konsistentes Image des Dateisystems auf Blockebene zu einem bestimmten Zeitpunkt liefert.

Dateisystemkonzepte

Bevor Sie Ihr Dateisystem verwalten und konfigurieren können, müssen Sie sich mit dem grundlegenden Aufbau und dem Inhalt des Dateibaums vertraut machen.

Organisation und Inhalt des Dateibaums:

Im Dateibaum werden Dateien mit ähnlichen Informationen in Verzeichnissen verwaltet. Diese Organisation vereinfacht das Anhängen von Verzeichnissen und Dateien über Remote-Zugriff.

Systemadministratoren können diese Verzeichnisse als Bausteine verwenden, um einen eindeutigen Dateibaum für jeden Client zu erstellen, der einzelne Verzeichnisse von einem oder mehreren Servern anhängt. Das Anhängen von Dateien und Verzeichnissen über Remote-Zugriff hat im Vergleich mit der lokalen Aufbewahrung von Informationen die folgenden Vorteile:

- Es wird Plattenspeicherplatz eingespart.
- Die Systemverwaltung ist einfach und zentral.
- Die Umgebung ist sicherer.

Der Dateibaum hat folgende Merkmale:

- Dateien, die von mehreren Maschinen derselben Hardwarearchitektur gemeinsam benutzt werden können, befinden sich im Dateisystem `/usr`.
- Variable Clientdateien, z. B. SPOOL- und Mail-Dateien, befinden sich im Dateisystem `/var`.
- Architekturunabhängige, gemeinsam nutzbare Textdateien, z. B. Handbuchseiten, befinden sich im Verzeichnis `/usr/share`.

- Das Stammdateisystem (/) enthält Dateien und Verzeichnisse, die für den Systembetrieb entscheidend sind. Beispielsweise enthält das Stammdateisystem ein Einheitenverzeichnis, Programme, die für den Systemstart verwendet werden, und Mountpunkte, über die Dateisysteme an das Stammdateisystem angehängt werden können.
- Das Dateisystem /home ist der Mountpunkt für die Ausgangsverzeichnisse von Benutzern.

Dateisystemstruktur:

Es ist wichtig, den Unterschied zwischen einem Dateisystem und einem Verzeichnis zu begreifen. Ein Dateisystem ist ein Abschnitt der Festplatte, der für Dateien reserviert ist. Für den Zugriff auf diesen Abschnitt der Festplatte wird das Dateisystem über ein Verzeichnis angehängt. Nach dem Anhängen erscheint das Dateisystem dem Benutzer wie jedes andere Verzeichnis.

Aufgrund der strukturellen Unterschiede zwischen Dateisystemen und Verzeichnissen können die Daten in diesen Entitäten jedoch separat verwaltet werden.

Bei einer Erstinstallation wird das Betriebssystem, wie in der folgenden Abbildung gezeigt, in eine Verzeichnisstruktur geladen.

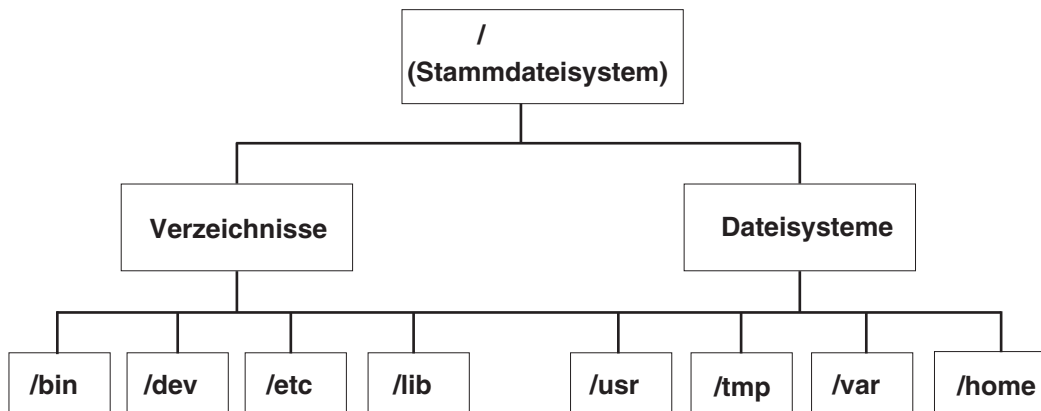


Abbildung 7. Baumstruktur des Dateisystems / (Stammdateisystem). Dieses Baumdiagramm zeigt eine Verzeichnisstruktur mit dem Stammdateisystem (/) ganz oben und von dort ausgehenden Verzweigungen zu Verzeichnissen und Dateisystemen. Verzeichnisse verzweigen zu /bin, /dev, /etc und /lib. Dateisysteme verzweigen zu /usr, /tmp, /var und /home.

Die Verzeichnisse auf der rechten Seite (/usr, /tmp, /var und /home) sind alles Dateisysteme. Deshalb sind ihnen separate Abschnitte auf der Festplatte zugeordnet. Diese Dateisysteme werden beim Systemstart automatisch angehängt, so dass der Endbenutzer den Unterschied zwischen diesen Dateisystemen und den auf der linken Seite aufgelisteten Verzeichnissen (/bin, /dev, /etc und /lib) nicht sieht.

Auf unabhängigen Maschinen sind die folgenden Dateisysteme standardmäßig auf der zugeordneten Einheit gespeichert:

| /Einheit | /Dateisystem |
|--------------|----------------------|
| /dev/hd1 | /home |
| /dev/hd2 | /usr |
| /dev/hd3 | /tmp |
| /dev/hd4 | / (Stammdateisystem) |
| /dev/hd9var | /var |
| /proc | /proc |
| /dev/hd10opt | /opt |

Der Dateibaum hat folgende Merkmale:

- Dateien, die von mehreren Maschinen derselben Hardwarearchitektur gemeinsam benutzt werden können, befinden sich im Dateisystem /usr.
- Variable Clientdateien, z. B. SPOOL- und Mail-Dateien, befinden sich im Dateisystem /var.
- Das Stammdateisystem (/) enthält die für den Systembetrieb kritischen Dateien und Verzeichnisse. Es enthält beispielsweise
 - ein Einheitenverzeichnis (/dev),
 - Mountpunkte, über die Dateisysteme an das Stammdateisystem angehängt werden können, z. B. /mnt.
- Das Dateisystem /home ist der Mountpunkt für die Ausgangsverzeichnisse der Benutzer.
- Für Server enthält das Verzeichnis /export die Paging-Bereichsdateien, die (nicht gemeinsam genutzten) Stammdateisysteme der einzelnen Clients, Speicherauszugs- und Ausgangsverzeichnisse und Verzeichnisse /usr/share für Clients ohne Plattenspeicher sowie exportierte Verzeichnisse /usr.
- Das Dateisystem /proc enthält Informationen zum Status der Prozesse und Threads im System.
- Das Dateisystem /opt enthält optionale Software, z. B. Anwendungen.

Die folgende Liste enthält Informationen zum Inhalt einiger Unterverzeichnisse des Stammdateisystems /.

| Eintrag | Beschreibung |
|---------|---|
| /bin | Symbolische Verbindung zum Verzeichnis /usr/bin. |
| /dev | Enthält Einheitenknoten für Gerätedateien für lokale Einheiten. Das Verzeichnis /dev enthält Gerätedateien für Bandlaufwerke, Drucker, Plattenpartitionen und Terminals. |
| /etc | Enthält Konfigurationsdateien, die für jede Maschine unterschiedlich sind. Beispiele: <ul style="list-style-type: none"> • /etc/hosts • /etc/passwd |
| /export | Enthält die Verzeichnisse und Dateien auf einem Server, die für ferne Clients bestimmt sind. |
| /home | Dient als Mountpunkt für ein Dateisystem, das Ausgangsverzeichnisse für Benutzer enthält. Das Dateisystem /home enthält benutzerspezifische Dateien und Verzeichnisse.

Auf einer eigenständigen Maschine wird ein gesondertes lokales Dateisystem über das Verzeichnis /home angehängt. In einem Netz kann ein Server Benutzerdateien enthalten, die für mehrere Maschinen zugänglich sein müssen. In diesem Fall wird die Serverkopie des Verzeichnisses /home fern an ein lokales Dateisystem /home angehängt. |
| /lib | Symbolische Verbindung zum Verzeichnis /usr/lib, das architekturabhängige Bibliotheken mit dem Namensformat lib*.a enthält. |
| /sbin | Enthält Dateien, die zum Booten der Maschine und zum Anhängen des Dateisystems /usr benötigt werden. Die meisten Befehle, die beim Booten verwendet werden, stammen aus dem RAM-Dateisystem des Boot-Image. Deshalb sind nur wenige Befehle im Verzeichnis /sbin gespeichert. |
| /tmp | Dient als Mountpunkt für ein Dateisystem, das vom System generierte temporäre Dateien enthält. |
| /u | Symbolische Verbindung zum Verzeichnis /home. |

| Eintrag | Beschreibung |
|---------|--|
| /usr | Dient als Mountpunkt für ein Dateisystem, das Dateien enthält, die sich nicht ändern und von mehreren Maschinen gemeinsam genutzt werden können (wie z. B. ausführbare Programme und ASCII-Dokumentation).

Eigenständige Maschinen hängen ein gesondertes lokales Dateisystem über das Verzeichnis /usr an. Maschinen ohne Plattenspeicher und Datenstationen mit systemreservierter kleiner Platte hängen ein Verzeichnis eines fernen Servers über das Dateisystem /usr an. |
| /var | Dient als Mountpunkt für Dateien, die auf jeder Maschine unterschiedlich sind. Das Dateisystem /var ist als Dateisystem konfiguriert, weil die darin enthaltenen Dateien dazu neigen, an Größe zuzunehmen. Beispielsweise ist es eine symbolische Verbindung zum Verzeichnis /usr/tmp, das temporäre Arbeitsdateien enthält. |

Stammdateisystem:

Das Stammdateisystem befindet sich ganz oben im hierarchischen Dateibaum. Es enthält die Dateien und Verzeichnisse, die für den Systembetrieb entscheidend sind. Dazu gehören das Einheitenverzeichnis und die Programme für das Booten des Systems. Das Stammdateisystem enthält außerdem die Mountpunkte, über die Dateisysteme angehängt werden können, um eine Verbindung zur Hierarchie des Stammdateisystems herzustellen.

Die folgende Abbildung zeigt eine Reihe von Unterverzeichnissen des Stammdateisystems.

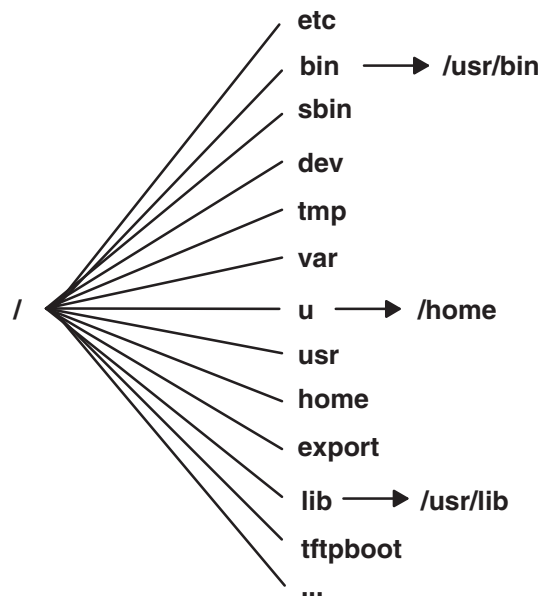


Abbildung 8. Stammdateisystem. Diese Abbildung zeigt das Stammdateisystem und seine Unterverzeichnisse. Das Unterverzeichnis /bin zeigt auf das Verzeichnis /usr/bin. Das Unterverzeichnis /lib zeigt auf das Verzeichnis /usr/lib. Das Unterverzeichnis /u zeigt auf das Verzeichnis /home.

Die folgende Liste enthält Informationen zum Inhalt einiger Unterverzeichnisse im Stammdateisystem (/).

| Eintrag | Beschreibung |
|----------------|--|
| /etc | <p>Enthält Konfigurationsdateien, die für jede Maschine unterschiedlich sind. Beispiele:</p> <ul style="list-style-type: none"> • /etc/hosts • /etc/passwd <p>Das Verzeichnis /etc enthält Dateien, die im Allgemeinen für die Systemverwaltung verwendet werden. Die meisten Befehle, die sich zuvor im Verzeichnis /etc befunden haben, befinden sich jetzt im Verzeichnis /usr/sbin. Aus Gründen der Kompatibilität enthält das Verzeichnis /usr/sbin jedoch symbolische Verbindungen zu den Positionen einiger ausführbarer Dateien. Beispiele:</p> <ul style="list-style-type: none"> • /etc/chown ist eine symbolische Verbindung zu /usr/bin/chown. • /etc/exportvg ist eine symbolische Verbindung zu /usr/sbin/exportvg. |
| /bin | Symbolische Verbindung zum Verzeichnis /usr/bin. In früheren UNIX-Dateisystemen enthielt das Verzeichnis /bin Benutzerbefehle, die sich jetzt im Verzeichnis /usr/bin befinden. |
| /sbin | Enthält Dateien, die zum Booten der Maschine und zum Anhängen des Dateisystems /usr benötigt werden. Die meisten Befehle, die beim Booten verwendet werden, stammen aus dem RAM-Dateisystem des Boot-Image. Deshalb sind nur wenige Befehle im Verzeichnis /sbin gespeichert. |
| /dev | Enthält Einheitenknoten für Gerätedateien für lokale Einheiten. Das Verzeichnis /dev enthält Gerätedateien für Bandlaufwerke, Drucker, Plattenpartitionen und Terminals. |
| /tmp | Dient als Mountpunkt für ein Dateisystem, das vom System generierte temporäre Dateien enthält. Das Dateisystem /tmp ist ein leeres Verzeichnis. |
| /var | Dient als Mountpunkt für Dateien, die auf jeder Maschine unterschiedlich sind. Das Dateisystem /var ist als Dateisystem konfiguriert, weil die darin enthaltenen Dateien dazu neigen, an Größe zuzunehmen. Weitere Informationen finden Sie im Abschnitt „Dateisystem /var“ auf Seite 461. |
| /u | Symbolische Verbindung zum Verzeichnis /home. |
| /usr | Enthält Dateien, die sich nicht ändern und von Maschinen gemeinsam genutzt werden können, wie z. B. ausführbare Dateien und ASCII-Dokumentation. |
| /home | <p>Eigenständige Maschinen hängen das Stammverzeichnis eines separaten lokalen Dateisystems über das Verzeichnis /usr an. Maschinen ohne Plattenspeicher und Maschinen mit eingeschränkten Plattenressourcen hängen ein Verzeichnis von einem fernen Server über das Dateisystem /usr an. Informationen zum Dateibaum, der über das Verzeichnis /usr angehängt wird, finden Sie im Abschnitt „Dateisystem /usr“.</p> <p>Dient als Mountpunkt für ein Dateisystem, das Ausgangsverzeichnisse für Benutzer enthält. Das Dateisystem /home enthält benutzerspezifische Dateien und Verzeichnisse.</p> <p>Auf einer eigenständigen Maschine befindet sich das Verzeichnis /home in einem separaten Dateisystem, dessen Stammverzeichnis über das Verzeichnis /home des Stammdateisystems angehängt wird. In einem Netz kann ein Server Benutzerdateien enthalten, die über mehrere Maschinen zugänglich sind. In diesem Fall wird die Serverkopie des Verzeichnisses /home über Remote-Zugriff an ein lokales Dateisystem /home angehängt.</p> |
| /export | Enthält die Verzeichnisse und Dateien auf einem Server, die für ferne Clients bestimmt sind. |
| /lib | Weitere Informationen zum Dateibaum im Verzeichnis /export finden Sie im Abschnitt „Verzeichnis /export“ auf Seite 461. |
| /tftpboot | Symbolische Verbindung zum Verzeichnis /usr/lib. Weitere Informationen finden Sie im Abschnitt „Dateisystem /usr“. |
| /tftpboot | Enthält Boot-Images und Bootinformationen für Clients ohne Plattenspeicher. |

Dateisystem /usr:

Das Dateisystem /usr enthält ausführbare Dateien, die von mehreren Maschinen gemeinsam genutzt werden können.

Die wichtigsten Unterverzeichnisse des Verzeichnisses /usr werden in der folgenden Abbildung gezeigt.

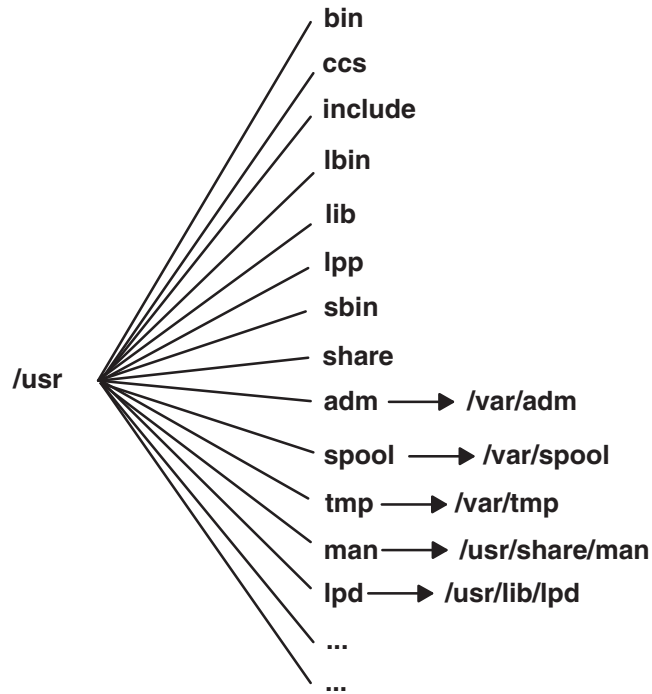


Abbildung 9. Dateisystem /usr. Diese Abbildung zeigt die wichtigsten Unterverzeichnisse des Verzeichnisses /usr, zu denen /bin, /ccs, /lib, /lpp, /adm und dessen Unterverzeichnis /var/adm sowie /man und dessen Unterverzeichnis /usr/share/man gehören.

Auf einer eigenständigen Maschine ist das Dateisystem /usr ein separates Dateisystem (auf dem logischen Datenträger /dev/hd2). Auf einer Maschine ohne Plattenpeicher oder auf einer Maschine mit eingeschränkten Plattenressourcen wird ein Verzeichnis von einem fernen Server mit Leseberechtigungen über das lokale Dateisystem /usr angehängt. Das Dateisystem /usr enthält schreibgeschützte Befehle, Bibliotheken und Daten.

Mit Ausnahme des Inhalts des Verzeichnisses /usr/share können die Dateien und Verzeichnisse im Dateisystem /usr von allen Maschinen derselben Hardwarearchitektur gemeinsam genutzt werden.

Das Dateisystem /usr enthält die folgenden Verzeichnisse:

| Eintrag | Beschreibung |
|--------------|--|
| /usr/bin | Enthält gewöhnliche Befehle und Shell-Scripts. Das Verzeichnis /usr/bin enthält beispielsweise die Befehle ls , cat und mkdir . |
| /usr/ccs | Enthält nicht gepackte Binärdateien aus Entwicklungspaketen. |
| /usr/include | Enthält Include- oder Header-Dateien. |
| /usr/lbin | Enthält ausführbare Dateien, die Back-Ends für Befehle sind. |
| /usr/lib | Enthält architekturunabhängige Bibliotheken mit Namen im Format lib*.a. Das Verzeichnis /lib im Stammverzeichnis (/) ist eine symbolische Verbindung zum Verzeichnis /usr/lib. Deshalb befinden sich jetzt alle Dateien, die im Verzeichnis /lib enthalten waren, im Verzeichnis /usr/lib. Dazu gehören aus Kompatibilitätsgründen auch einige Dateien, die keine Bibliotheksdateien sind. |
| /usr/lpp | Enthält optional installierte Produkte. |
| /usr/sbin | Enthält Dienstprogramme, die für die Systemverwaltung verwendet werden, z. B. SMIT-Befehle (System Management Interface Tool). Die meisten Befehle, die früher im Verzeichnis /etc abgelegt waren, befinden sich jetzt im Verzeichnis /usr/sbin. |
| /usr/share | Enthält Dateien, die von Maschinen mit unterschiedlichen Architekturen gemeinsam genutzt werden können. Weitere Informationen finden Sie im Abschnitt „Verzeichnis /usr/share“ auf Seite 460. |

Im Folgenden sind symbolische Verbindungen zum Verzeichnis /var aufgelistet:

| Eintrag | Beschreibung |
|---------------|---|
| /usr/adm | Symbolische Verbindung zum Verzeichnis /var/adm. |
| /usr/mail | Symbolische Verbindung zum Verzeichnis /var/spool/mail. |
| /usr/news | Symbolische Verbindung zum Verzeichnis /var/news. |
| /usr/preserve | Symbolische Verbindung zum Verzeichnis /var/preserve. |
| /usr/spool | Symbolische Verbindung zum Verzeichnis /var/spool. |
| /usr/tmp | Symbolische Verbindung zum Verzeichnis /var/tmp, weil das Verzeichnis /usr potenziell von vielen Knoten gemeinsam genutzt werden kann und schreibgeschützt ist. |

Im Folgenden sind symbolische Verbindungen zu den Verzeichnissen /usr/share und /usr/lib aufgelistet:

| Eintrag | Beschreibung |
|-----------|---|
| /usr/dict | Symbolische Verbindung zum Verzeichnis /usr/share/dict. |
| /usr/man | Symbolische Verbindung zum Verzeichnis /usr/share/man. |
| /usr/lpd | Symbolische Verbindung zum Verzeichnis /usr/lib/lpd. |

Verzeichnis /usr/share:

Das Verzeichnis /usr/share enthält architekturunabhängige gemeinsam nutzbare Textdateien. Der Inhalt dieses Verzeichnisses kann von allen Maschinen, unabhängig von der Hardwarearchitektur gemeinsam genutzt werden.

In einer Umgebung mit unterschiedlichen Architekturen hängt der typische Client ohne Plattenspeicher ein Serververzeichnis über sein eigenes Verzeichnis /usr und anschließend ein anderes Verzeichnis über das Verzeichnis /usr/share an. Die Dateien unterhalb des Verzeichnisses /usr/share sind in einem oder mehreren gesondert installierbaren Paketen enthalten. Auf einem Knoten können somit andere Teile des Verzeichnisses /usr, von denen er abhängig ist, lokal installiert sein, während ein Server verwendet wird, der das Verzeichnis /usr/share bereitstellt.

Zu den Dateien im Verzeichnis /usr/share gehören die Verzeichnisse und Dateien, die in der folgenden Abbildung gezeigt werden.

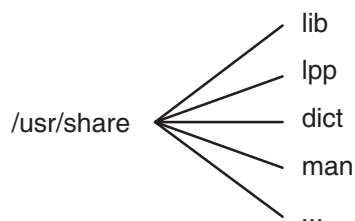


Abbildung 10. Verzeichnis /usr/share.

Diese Abbildung zeigt verschiedene Verzeichnisse unterhalb des Verzeichnisses /usr/share, darunter /lib, /lpp, /dict und /man.

Das Verzeichnis /usr/share enthält die folgenden Elemente:

| Eintrag | Beschreibung |
|-----------------|--|
| /usr/share/man | Enthält die Man-Pages, sofern diese geladen wurden. |
| /usr/share/dict | Enthält das Wörterverzeichnis für Rechtschreibprüfung und die zugehörigen Indizes. |
| /usr/share/lib | Enthält architekturunabhängige Datendateien, z. B. terminfo, learn, tmac, me und macros. |
| /usr/share/lpp | Enthält Daten und Informationen zu optional installierbaren Produkten auf dem System. |

Dateisystem /var:

Das Dateisystem /var neigt zum Wachstum, weil es Unterverzeichnisse und Datendateien enthält, die von viel genutzten Anwendungen, wie z. B. Abrechnung, Mail und dem Druckspooler, verwendet werden.

Achtung: Wenn Anwendungen auf Ihrem System das Dateisystem /var ausgiebig verwenden, sollten Sie in regelmäßigen Abständen den Befehl **skulker** ausführen oder eine höhere Dateisystemgröße als den Standardwert von 4 MB für /var festlegen.

Spezielle Dateien im Dateisystem /var, die regelmäßig überwacht werden sollten, sind /var/adm/wtmp und /var/adm/ras/errlog.

Auch die folgenden Dateien in /var sollten überwacht werden:

| Eintrag | Beschreibung |
|----------------------|--|
| /var/adm/ras/trcfile | Wenn die Tracefunktion aktiviert ist. |
| /var/tmp/snmpd.log | Wenn der Befehl snmpd auf dem System ausgeführt wird. |

Die Abbildung "Verzeichnis /var" zeigt einige Verzeichnisse im Dateisystem /var.

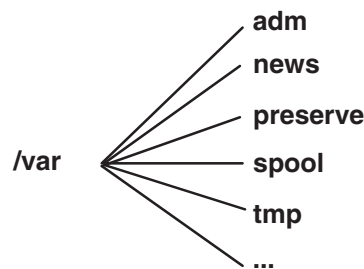


Abbildung 11. Verzeichnis /var. Diese Abbildung zeigt die wichtigsten Unterverzeichnisse im Verzeichnis /var, z. B. /adm, /news, /preserve, /spool und /tmp.

| Eintrag | Beschreibung |
|----------------|---|
| /var/adm | Enthält Dateien zur Systemprotokollierung und -abrechnung. |
| /var/news | Enthält Systemnachrichten. |
| /var/preserve | Enthält aufgehobene Daten aus unterbrochenen Editiersitzungen. Gleicht dem Verzeichnis /usr/preserve in früheren Releases. |
| /var/spool | Enthält Dateien, die von Programmen wie E-Mail verarbeitet werden. Gleicht dem Verzeichnis /usr/spool in früheren Releases. |
| /var/tmp | Enthält temporäre Dateien. Gleicht dem Verzeichnis /usr/tmp in früheren Releases. Das Verzeichnis /usr/tmp ist jetzt eine symbolische Verbindung zu /var/tmp. |

Verzeichnis /export:

Das Verzeichnis /export enthält Serverdateien, die auf Clients, wie z. B. Workstations ohne Plattenspeicher, Workstations mit systemreservierter Platte oder Workstations mit systemreservierter kleiner Platte, exportiert werden.

Ein Server kann verschiedene Typen von Plattenspeicherplatz exportieren. Dazu gehören auch Pakete mit ausführbaren Programmen, Paging-Bereich und Stammdateisysteme für Clients ohne Plattenspeicher oder Clients mit wenig Plattenressourcen. Die Standardposition für solchen Plattenspeicherplatz im Dateibaum ist das Verzeichnis `/export`. Einige Unterverzeichnisse des Verzeichnisses `/export` sind in der folgenden Liste beschrieben:

- /exec** Enthält Verzeichnisse, die Clients ohne Plattenspeicher über ihre Dateisysteme `/usr` anhängen.
- /swap** Enthält Dateien für fernes Paging von Clients ohne Plattenspeicher.
- /share** Enthält Verzeichnisse, die Clients ohne Plattenspeicher über ihr Verzeichnis `/usr/share` anhängen.
- /root** Enthält Verzeichnisse, die Clients ohne Plattenspeicher über ihr Stammdateisystem `/` anhängen.
- /home** Enthält Verzeichnisse, die Clients ohne Plattenspeicher über ihr Dateisystem `/home` anhängen.

Das Verzeichnis `/export` ist die Standardposition für Clientressourcen, die von den Befehlen für Workstations ohne Plattenspeicher bearbeitet werden. Das Verzeichnis `/export` enthält als einziges Clientressourcen auf dem Server. Da Clients diese Ressourcen an ihren eigenen Dateibaum anhängen, scheinen sich diese Ressourcen für die Clients an den "normalen" Stellen in einem Dateibaum zu befinden. Im Folgenden sind die wichtigsten Unterverzeichnisse des Verzeichnisses `/export` und ihre Mountpunkte in einem Clientdateibaum beschrieben:

/export/root

Dieses Verzeichnis wird über das Stammdateisystem des Clients (`/`) angehängt. Clientstammverzeichnisse befinden sich standardmäßig im Verzeichnis `/export/root` und sind nach dem Hostnamen des Clients benannt.

/export/exec

Dieses Verzeichnis wird auch als SPOT-Verzeichnis (Shared Product Object Tree) bezeichnet. Es wird über das Clientdateisystem `/usr` angehängt. SPOTs sind Versionen des Dateisystems `/usr`, die im Verzeichnis `/export/exec` gespeichert sind, und haben Namen, die das Release-Level angeben. Standardmäßig lautet der Name `RISCAIX`.

/export/share

Dieses Verzeichnis wird über das Clientverzeichnis `/usr/share` angehängt. Dieses Verzeichnis enthält Daten, die von vielen Architekturen gemeinsam genutzt werden können. Die Standardposition ist `/export/share/AIX/usr/share`.

/export/home

Dieses Verzeichnis wird über das Clientdateisystem `/home` angehängt. Es enthält Benutzerverzeichnisse, die nach Clienthostnamen gruppiert sind. Die Standardposition für die Clientausgangsverzeichnisse ist `/export/home`.

/export/swap

Dieses Verzeichnis wird auch Paging-Verzeichnis genannt. Auf eigenständigen Systemen und Systemen mit systemreservierter Platte wird das Paging von einer lokalen Platte unterstützt. Dieser Service wird bei Clients ohne Plattenspeicher von einer Datei auf einem Server bereitgestellt. Die Datei ist nach dem Hostnamen des Clients benannt und befindet sich standardmäßig im Verzeichnis `/export/swap`.

/export/dump

Eigenständige Systeme verwenden eine lokale Platte als Speicherauszugseinheit. Clients ohne Plattenspeicher verwenden eine Datei auf dem Server. Die Datei ist in einem Verzeichnis enthalten, das nach dem Hostnamen des Clients benannt ist und sich standardmäßig im Verzeichnis `/export/dump` befindet.

microcode

Dieses Verzeichnis enthält Mikrocode für physische Einheiten. Die Standardposition ist `/export/exec/RISCAIX/usr/lib/microcode`.

JFS2-Dateisysteme verschlüsseln:

Beginnend mit AIX Version 6.1 wird EFS (Encrypted File System) auf JFS2-Dateisystemen unterstützt. EFS erlaubt die Verschlüsselung Ihrer Daten und die Steuerung des Datenzugriffs über einen verschlüsselten Zugriffsschutz.

Jedem Benutzer wird ein Schlüssel zugeordnet und in einem verschlüsselten Schlüsselspeicher gespeichert. Nach erfolgreicher Anmeldung werden die Schlüssel des Benutzers in den Kernel geladen und den Berechtigungsnachweisen für den Prozess zugeordnet. Beim Öffnen einer EFS-geschützten Datei werden die Berechtigungsnachweise für den Prozess geprüft. Wenn der Prozess einen Schlüssel findet, der mit dem für die Datei festgelegten Zugriffsschutz übereinstimmt, entschlüsselt er den Dateischlüssel und den Dateiinhalt.

Standardmäßig sind JFS2-Dateisysteme nicht für EFS aktiviert. Ein JFS2-Dateisystem muss für EFS aktiviert sein, damit Daten verschlüsselt werden können. Informationen dazu, wie EFS aktiviert wird, finden Sie in der Beschreibung des Befehls **efsenable** in der Veröffentlichung *Commands Reference, Volume 2*.

Dateisysteme konfigurieren

Wenn Sie Dateisysteme hinzufügen oder konfigurieren, können Sie Optionen im Container "Dateisysteme" der SMIT-Direktaufrufe verwenden.

Die SMIT-Direktaufrufe sind in der folgenden Tabelle aufgeführt:

Tabelle 64. Logische Datenträger verwalten und Dateisystemtasks

| <i>Task</i> | <i>SMIT-Direktaufruf</i> |
|---|--|
| JFS oder JFS2 hinzufügen | smit crfs |
| JFS2 zu einem vorhandenen logischen Datenträger hinzufügen | smit crjfs2lvstd |
| JFS zu einem zuvor definierten Menü für logische Datenträger hinzufügen | Logischen Datenträger erstellen und anschließend smit crjfslv ausführen |
| Attribute eines JFS oder JFS2 ändern <small>Anmerkung 1</small> | smit chfs |
| Größe eines Dateisystems prüfen | smit fs |
| Größe eines Dateisystems erhöhen | JFS: smit chjfs JFS2: smit chjfs2 |
| Größe eines Dateisystems verringern | JFS2: smit chjfs2 |

Anmerkung: Der SMIT-Direktaufruf für **Größe eines Dateisystems verringern** gilt nur für JFS2.

Dateisystem verwalten

Ein Dateisystem ist eine vollständige Verzeichnisstruktur mit einem Stammverzeichnis und allen untergeordneten Unterverzeichnissen und Dateien.

Dateisysteme sind auf einen einzigen logischen Datenträger beschränkt. Einige der wichtigsten Systemverwaltungstasks beziehen sich auf Dateisysteme. Dazu gehören insbesondere:

- Speicherplatz für Dateisysteme auf logischen Datenträgern reservieren
- Dateisysteme erstellen
- Systembenutzern Speicherplatz im Dateisystem zur Verfügung stellen
- Speicherplatzauslastung des Dateisystems überwachen
- Dateisysteme zum Schutz vor Datenverlusten bei Systemausfällen sichern
- Momentaufnahme erstellen, um ein konsistentes Image eines Dateisystems auf Blockebene zu einem bestimmten Zeitpunkt zu speichern
- Konsistenz in einem Dateisystem erhalten

In der folgenden Liste sind die Systemverwaltungsbefehle aufgeführt, mit denen Sie Dateisysteme verwalten können:

| Eintrag | Beschreibung |
|--------------------------|--|
| backup | Erstellt eine vollständige oder inkrementelle Sicherung eines Dateisystems. |
| chfs -a splitcopy | Erstellt eine Onlinesicherung eines angehängten JFS-Dateisystems. |
| dd | Kopiert für Dateisystemsicherungen Daten direkt von einer Einheit auf eine andere. |
| df | Meldet den belegten und freien Speicherplatz in einem Dateisystem zurück. |
| fsck | Überprüft Dateisysteme und korrigiert Inkonsistenzen. |
| mkfs | Erstellt ein Dateisystem in einer angegebenen Größe auf einem angegebenen logischen Datenträger. |
| mount | Hängt ein Dateisystem an die systemweite Namensstruktur an, so dass auf die Dateien und Verzeichnisse in diesem Dateisystem zugegriffen werden kann. |
| restore | Stellt Dateien aus einer Sicherung wieder her. |
| snapshot | Erstellt eine Momentaufnahme eines JFS2-Dateisystems. |
| umount | Entfernt ein Dateisystem aus der systemweiten Namensstruktur und macht damit die Dateien und Verzeichnisse in diesem Dateisystem unzugänglich. |

In einem Dateisystem verfügbaren Speicherplatz anzeigen (Befehl **df**):

Mit dem Befehl **df** können Sie Informationen zum gesamten und zum verfügbaren Speicherplatz in einem Dateisystem anzeigen. Der Parameter **Dateisystem** gibt den Namen der Einheit, auf dem sich das Dateisystem befindet, das Verzeichnis, an das das Verzeichnis angehängt ist, oder den Pfadnamen eines Dateisystems an.

Wenn Sie den Parameter **Dateisystem** nicht angeben, zeigt der Befehl **df** Informationen zu allen momentan angehängten Dateisystemen an. Wenn Sie eine Datei oder ein Verzeichnis angeben, zeigt der Befehl **df** Informationen zu dem Dateisystem an, in dem sich die Datei befindet.

In der Regel verwendet der Befehl **df** die Anzahl freier Blöcke im Superblock. In bestimmten Ausnahmefällen können diese Zähler fehlerhaft sein, beispielsweise wenn ein Dateisystem während der Ausführung des Befehls **df** aktiv geändert wird.

Die vollständige Syntax des Befehls **df** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

Anmerkung: In einigen fernen Dateisystemen, z. B. Network File Systems (NFS), bleiben die Spalten für den verfügbaren Speicherplatz am Bildschirm leer, wenn der Server diese Informationen nicht liefert.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **df**:

- Geben Sie beispielsweise Folgendes ein, um Informationen zu allen angehängten Dateisystemen anzuzeigen:

```
df
```

Falls Ihr System so konfiguriert ist, dass die Verzeichnisse `/`, `/usr`, `/site` und `/usr/venus` in gesonderten Dateisystemen abgelegt sind, sieht die Ausgabe des Befehls **df** in etwa wie folgt aus:

```
Filesystem 512-blocks  free   %used   Iused %Iused   Mounted on
/dev/hd4    20480    13780   32%     805   13%     /
/dev/hd2   385024    15772   95%    27715  28%    /usr
/dev/hd9var 40960    38988    4%     115    1%     /var
/dev/hd3    20480    18972    7%      81    1%     /tmp
/dev/hd1     4096     3724    9%      44    4%     /home
```

- Geben Sie Folgendes ein, um den verfügbaren Speicherplatz des Dateisystems anzuzeigen, in dem sich das aktuelle Verzeichnis befindet:

```
df .
```

Dateisystembefehle:

Es gibt eine Reihe von Befehlen, die für die Bearbeitung jedes Typs von Dateisystem verwendet werden können.

Die Datei `/etc/filesystems` steuert die Liste der Dateisysteme, die mit den folgenden Befehlen bearbeitet werden können:

| Eintrag | Beschreibung |
|--------------------|---|
| <code>chfs</code> | Ändert die Merkmale eines Dateisystems. |
| <code>crfs</code> | Fügt ein Dateisystem hinzu. |
| <code>lsfs</code> | Zeigt die Merkmale eines Dateisystems an. |
| <code>rmfs</code> | Entfernt ein Dateisystem. |
| <code>mount</code> | Stellt ein Dateisystem zur Verwendung bereit. |

Es gibt vier Befehle für die Bearbeitung virtueller Dateisystemtypen. Die Datei `/etc/vfs` enthält die Informationen zu den Dateisystemtypen, die mit den folgenden Befehlen bearbeitet werden können:

| Eintrag | Beschreibung |
|--------------------|--|
| <code>chvfs</code> | Ändert die Merkmale eines Dateisystemtyps. |
| <code>crvfs</code> | Fügt einen neuen Dateisystemtyp hinzu. |
| <code>lsvfs</code> | Listet die Merkmale eines Dateisystemtyps auf. |
| <code>rmvfs</code> | Entfernt einen Dateisystemtyp. |

Dateisysteme auf unterschiedlichen Maschinen vergleichen:

Wenn die auf den verschiedenen Maschinen vorhandenen Dateisysteme identisch sein sollen, Sie aber vermuten, dass eines davon beschädigt ist, können Sie die Dateisysteme vergleichen.

Die folgende Prozedur beschreibt, wie Sie die Attribute eines Dateisystems, das sich auf dem aktuellen Host (hier `orig_host`) befindet, mit demselben Dateisystem auf einem fernen Host vergleichen können.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Melden Sie sich als Root an dem fernen Host an. Beispiel:

```
tn juniper.mycompany.com
```

```
AIX Version 6.1
(C) Copyrights by IBM and by others 1982, 2002. login: root
root's Password:
```

2. Fügen Sie der Datei `.rhosts` des fernen Hosts in einem Editor eine Zeilengruppe hinzu, die Root die Ausführung sicherer ferner Befehle ermöglicht. Verwenden Sie für die neue Zeilengruppe das folgende Format:

```
orig_host root
```

Die geänderte Datei `.rhosts` kann wie folgt aussehen:

```
NIM.mycompany.com root
nim.mycompany.com root
host.othernetwork.com root
orig_host.mycompany.com root
```

3. Speichern Sie Ihre Änderungen und beenden Sie die Fernverbindung.
4. Erstellen Sie als Root auf `orig_host` in Ihrem Editor eine weitere Datei. In diesem Szenario hat die neue Datei den Namen `compareFS`. Beispiel:

```
vi compareFS
```

5. Fügen Sie der Datei den folgenden Text hinzu, in dem `FSname` für den Namen des Dateisystems steht, das Sie vergleichen möchten, und `remote_host` für den Namen des Hosts steht, auf dem sich das Vergleichsdateisystem befindet:

```
FSname -> remote_host
install -v ;
```

Anmerkung: In dieser Datei muss in der Zeile mit dem Befehl **install** zwischen dem Parameter **-v** und dem Semikolon (;) ein Leerzeichen stehen.

Beispiel:

```
/home/jane/* -> juniper.mycompany.com
install -v ;
```

6. Speichern Sie die Datei und beenden Sie den Editor. Die Datei `compareFS` wird im folgenden Schritt als `distfile` für den Befehl **rdist** verwendet.

7. Geben Sie an der Eingabeaufforderung Folgendes ein:

```
/usr/bin/rdist -f compareFS
```

Wenn Sie annehmen, dass die Ausgabe der Vergleichsoperation sehr umfangreich ist, senden Sie sie an eine Datei. Beispiel:

```
/usr/bin/rdist -f compareFS > compareFS_output
```

In der Ausgabe werden die Unterschiede zwischen den Dateisystemen aufgelistet.

Zugehörige Informationen:

rdist command

rhosts File Format for TCP/IP

Understanding the Secure Remote Commands

Dateisysteme verwalten

Die einfachsten Tasks, die Sie für die Verwaltung von Dateisystemen benötigen, sind in der folgenden Tabelle gruppiert.

Tabelle 65. Tasks für die Verwaltung von Dateisystemen

| Task | SMIT-Direktaufruf | Befehl oder Datei |
|--|----------------------------|---|
| Dateien oder Verzeichnisse nach Namen sichern | smit backfile | backup <small>Anmerkung 1</small> |
| Image einer JFS2-Momentaufnahme erstellen und sichern | smit backsnap | backsnap <small>Anmerkung</small> |
| Alle Dateisysteme auf einer Platte auflisten | smit lsmntdsk | |
| Dateisysteme auf einem austauschbaren Datenträger auflisten | smit lsmntdsk | |
| Angehängte Dateisysteme auflisten | smit fs | |
| Gruppe von Dateisystemen anhängen <small>Anmerkung 5</small> | smit mountg | mount -t <i>Gruppenname</i> |
| JFS oder JFS2 anhängen <small>Anmerkung 3</small> | smit mountfs | mount |
| JFS2-Momentaufnahme anhängen | smit mntsnap | mount -v jfs2 -o snapshot <i>Einheit Mountpunkt</i> |
| JFS oder JFS2 entfernen | smit rmfs | |
| JFS2-Momentaufnahme entfernen | smit rmsnap | snapshot -d <i>Einheit_mit_Momentaufnahme</i> |
| JFS2-Dateisystem auf den Zeitpunkt der Momentaufnahme zurücksetzen | smit rollbacksnap | rollback [-s] [-v] [-c] <i>snappedFS Objekt_mit_Momentaufnahme</i> |
| Dateisystem abhängen <small>Anmerkung</small> | smit umountfs | |
| Dateisystem auf einem austauschbaren Datenträger abhängen <small>Anmerkung 4</small> | smit umntdsk | |
| Gruppe von Dateisystemen abhängen <small>Anmerkung 5</small> | smit umountg | umount -t <i>Gruppenname</i> |
| JFS2-Kontingente (Enhanced Journaled File System) verwalten | smit j2fsquotas | |
| Kontingentverwaltung aktivieren und inaktivieren | smit j2enablequotas | |

Tabelle 65. Tasks für die Verwaltung von Dateisystemen (Forts.)

| Task | SMIT-Direktaufruf | Befehl oder Datei |
|---|-----------------------------|--------------------------|
| Durchsetzung von Kontingentbegrenzungen stoppen/erneut starten | smit j2enforcequotas | quotaon off -v |
| Kontingentnutzung auflisten | smit j2repquota | repquota -v |
| Statistiken zur aktuellen Nutzung von Plattenblöcken und Dateien erneut berechnen | smit j2quotacheck | quotacheck -v |
| Klasse limits hinzufügen | smit j2addlimit | j2edlimit -e |
| Merkmale einer Klasse limits ändern/anzeigen | smit j2changelimit | |
| Klasse limits als Standardgrenzwerte für ein Dateisystem festlegen | smit j2defaultlimit | |
| Benutzer oder Gruppe zu einer Klasse limits zuordnen | smit j2assignlimit | |
| Klassen limits für ein Dateisystem auflisten | smit j2listlimits | j2edlimit -l '-u' |
| Klasse limits entfernen | smit j2removelimit | |

Anmerkung:

1. Informationen zu den Optionen können Sie den Beschreibungen der einzelnen Befehle entnehmen.
2. Ändern Sie nicht die Namen von systemkritischen Dateisystemen. Es handelt sich dabei um / (Stammdateisystem) auf dem logischen Datenträger 4 (hd4), /usr auf hd2, /var auf hd9var, /tmp auf hd3 und /b1v auf hd5. Wenn Sie die Konvention *hdn* verwenden, beginnen Sie mit *hd10*.
3. Überprüfen Sie die Dateisysteme vor dem Anhängen mit der im Abschnitt „Dateisystem überprüfen“ auf Seite 470 beschriebenen Prozedur oder mit dem Befehl **fsck**.
4. Wenn das Abhängen eines Dateisystems scheitert, kann dies darauf zurückzuführen sein, dass ein Benutzer oder Prozess eine Datei im abzuhängenden Dateisystem geöffnet und diese noch im Zugriff hat. Mit dem Befehl **fuser** können Sie feststellen, welcher Benutzer oder Prozess den Fehler verursacht.
5. Eine Dateisystemgruppe ist eine Sammlung von Dateisystemen, die denselben Wert für die Kennung **type=** in der Datei */etc/filesystems* haben.

Zugehörige Tasks:

„Speicherplatz von einem anderen Plattenlaufwerk anhängen“ auf Seite 398

Sie können mehr Speicherplatz auf einem Plattenlaufwerk anfordern, indem Sie Speicherplatz von einem anderen Laufwerk anhängen.

Eine oder mehrere Dateien aus einer online verfügbaren externen JFS2-Momentaufnahme wiederherstellen:

Wenn eine Datei beschädigt ist, können Sie sie ersetzen, sofern Sie eine genaue Kopie in einer online verfügbaren externen JFS2-Momentaufnahme besitzen.

Gehen Sie wie folgt vor, um eine oder mehrere Dateien aus dem Image einer online verfügbaren externen JFS2-Momentaufnahme wiederherzustellen.

In diesem Beispiel wird davon ausgegangen, dass */home/aaa/meineDatei* eine beschädigte Datei im Dateisystem */home* ist.

1. Hängen Sie die Momentaufnahme mit einem ähnlichen Befehl wie dem folgenden an:

```
mount -v jfs2 -o snapshot /dev/mysnaplv /tmp/mysnap
```

2. Wechseln Sie in das Verzeichnis, in dem die Momentaufnahme enthalten ist, indem Sie einen Befehl ähnlich dem folgenden ausführen:

```
cd /tmp/mysnap
```

3. Kopieren Sie die genaue Kopie der Datei aus der Momentaufnahme, um die beschädigte Datei zu überschreiben, indem Sie einen Befehl ähnlich dem folgenden ausführen:

```
cp aaa/meineDatei /home/aaa/meineDatei
```

Dieser Befehl kopiert nur die Datei mit dem Namen `meineDatei`. Wenn Sie alle Dateien aus der Momentaufnahme in das Verzeichnis `aaa` kopieren möchten, geben Sie einen ähnlichen Befehl wie den folgenden ein:

```
cp -R aaa /home/aaa
```

Weitere Beispiele zum Ersetzen von beschädigten Dateien durch Images von Momentaufnahmen finden Sie in den Beschreibungen der Befehle `cp` und `cpio` in der Veröffentlichung *Commands Reference, Volume 1*.

Eine oder mehrere Dateien aus einer online verfügbaren internen JFS2-Momentaufnahme wiederherstellen:

Wenn eine Datei beschädigt ist, können Sie sie ersetzen, sofern Sie eine genaue Kopie in einer online verfügbaren internen JFS2-Momentaufnahme besitzen.

Gehen Sie wie folgt vor, um eine oder mehrere Dateien aus dem Image einer online verfügbaren internen JFS2-Momentaufnahme wiederherzustellen.

In diesem Beispiel wird davon ausgegangen, dass `/home/aaa/meineDatei` eine beschädigte Datei im Dateisystem `/home` ist.

1. Wechseln Sie in das Verzeichnis, in dem die Momentaufnahme enthalten ist, indem Sie einen Befehl ähnlich dem folgenden ausführen:

```
cd /home/.snapshot/mysnap
```

2. Kopieren Sie die genaue Kopie der Datei aus der Momentaufnahme, um die beschädigte Datei zu überschreiben, indem Sie einen Befehl ähnlich dem folgenden ausführen:

```
cp aaa/meineDatei /home/aaa/meineDatei
```

Dieser Befehl kopiert nur die Datei mit dem Namen `meineDatei`. Wenn Sie alle Dateien aus der Momentaufnahme in das Verzeichnis `aaa` kopieren möchten, geben Sie einen ähnlichen Befehl wie den folgenden ein:

```
cp -R aaa /home/aaa
```

Weitere Beispiele zum Ersetzen von beschädigten Dateien durch Images von Momentaufnahmen finden Sie in den Beschreibungen der Befehle `cp` und `cpio` in der Veröffentlichung *Commands Reference, Volume 1*.

Dateisysteme auf CD-ROMs und DVDs:

CDs und DVDs werden nicht automatisch angehängt, aber diese Funktion kann aktiviert werden.

Verwenden Sie zum Aktivieren dieser Funktion den Befehl `cdmount`, um das CD-ROM- (CDRFS) oder DVD-ROM-Dateisystem (UDFS) anzuhängen. Beispiel:

```
cdmount cd0
```

Mit dem folgenden Befehl können Sie ein DVD-ROM-Dateisystem im Lese-/Schreibzugriff manuell anhängen:

```
mount -V udfs Einheitennamen Mountpunkt
```

Einheitenname steht für den Namen des DVD-Laufwerks und *Mountpunkt* für den Mountpunkt für das Dateisystem.

Dateisysteme auf wieder beschreibbaren optischen Datenträgern verwenden:

CDRFS- (CR-ROM File System) und JFS-Dateisysteme können auf wieder beschreibbaren optischen Datenträgern verwendet werden.

Ein CD-ROM-Dateisystem (CDRFS) kann auf einem wieder beschreibbaren optischen Datenträger (sofern dieser schreibgeschützt ist) und auf einer CD-ROM gespeichert werden. In der folgenden Tabelle wird beschrieben, wie Sie ein CDRFS auf einem wieder beschreibbaren optischen Datenträger hinzufügen, anhängen und abhängen. Sie müssen die folgenden Informationen angeben, wenn Sie das Dateisystem anhängen:

| | |
|-------------------------------|---|
| Eintrag | Beschreibung |
| Einheitenname | Definiert den Namen der Einheit, die den Datenträger enthält. |
| Mountpunkt | Gibt das Verzeichnis an, über das das Dateisystem angehängt wird. |
| Automatisches Anhängen | Gibt an, ob das Dateisystem automatisch beim Systemwiederanlauf angehängt wird. |

| Tasks für CDRFS auf optischen Datenträgern | | |
|--|--|---|
| Task | SMIT-Direktaufruf | Befehl oder Datei |
| CDRFS hinzufügen ¹ | smit crcdrfs | 1. Dateisystem hinzufügen: crfs -v cdrfs -p ro -dEinheitenname -m Mountpunkt -A Automatisches Anhängen
2. Dateisystem anhängen: mount Mountpunkt |
| CDRFS entfernen ² | 1. Dateisystem abhängen: smit umountfs
2. Dateisystem entfernen: smit rmcdrfs | 1. Dateisystem abhängen: umount Dateisystem
2. Dateisystem entfernen: rdfs Mountpunkt |

Anmerkung:

- Stellen Sie sicher, dass der wieder beschreibbare optische Datenträger schreibgeschützt ist.
- Ein CDRFS-Dateisystem muss abgehängt werden, bevor der wieder beschreibbare optische Datenträger entfernt werden kann.

Ein JFS stellt auf einem optischen Datenträger ein ähnliches Dateisystem mit Schreib-/Lesezugriff wie auf einer Festplatte bereit. Sie müssen Systemberechtigung besitzen, um ein Dateisystem mit Schreib-/Lesezugriff auf einem wieder beschreibbaren optischen Datenträger zu erstellen oder zu importieren (d. h., Ihre Anmeldung muss zur Gruppe system gehören). Außerdem benötigen Sie die folgenden Informationen:

Name der Datenträgergruppe

Gibt den Namen der Datenträgergruppe an.

Einheitenname

Gibt den logischen Namen des Laufwerks für wieder beschreibbare optische Platten an.

Mountpunkt

Gibt die Verzeichnisse an, über die Dateisysteme angehängt werden.

Automatisches Anhängen

Gibt an, ob das Dateisystem automatisch beim Systemwiederanlauf angehängt wird.

Anmerkung:

- Alle Datenträgergruppen, die auf dem wieder beschreibbaren optischen Datenträger erstellt werden, müssen vollständig auf diesem Datenträger enthalten sein. Datenträgergruppen können sich nicht über die Grenzen einer wieder beschreibbaren optischen Platte hinaus erstrecken.
- Wenn Sie auf ein zuvor erstelltes JFS zugreifen, muss der Name der Datenträgergruppe nicht mit dem Namen übereinstimmen, der beim Erstellen der Datengruppe verwendet wurde.

| Tasks für JFS auf optischen Datenträgern | | |
|---|--|---|
| Task | SMIT-Direktaufruf | Befehl oder Datei |
| JFS hinzufügen | <ol style="list-style-type: none"> 1. Optische Platte in das Laufwerk einlegen. 2. Datenträgergruppe erstellen (sofern erforderlich): smit mkvg 3. Journaled File System erstellen: smit crfs | <ol style="list-style-type: none"> 1. Optische Platte in das Laufwerk einlegen. 2. Datenträgergruppe erstellen (sofern erforderlich): mkvg -f -y
<i>Name_der_Datenträgergruppe -d 1</i>
<i>Einheitenname</i> 3. Journaled File System erstellen: crfs -v jfs -g Name_der_Datenträgergruppe -a size=Größe_des_Dateisystems -m
<i>Mountpunkt -A Automatisches Anhängen</i>
-p rw 4. Dateisystem anhängen: mount
<i>Mountpunkt</i> |
| Auf ein zuvor erstelltes JFS zugreifen <small>Anmerkung 1</small> | <ol style="list-style-type: none"> 1. Optische Platte in das Laufwerk einlegen. 2. Datenträgergruppe importieren: smit importvg | <ol style="list-style-type: none"> 1. Optische Platte in das Laufwerk einlegen. 2. Datenträgergruppe importieren: importvg -y
<i>Name_der_Datenträgergruppe</i>
<i>Einheitenname</i> 3. Dateisystem anhängen: mount
<i>Mountpunkt</i> |
| JFS entfernen <small>Anmerkung 2</small> | <ol style="list-style-type: none"> 1. Dateisystem abhängen: smit umountfs 2. Dateisystem entfernen: smit rmjfs | <ol style="list-style-type: none"> 1. Dateisystem abhängen: umount
<i>Dateisystem</i> 2. Dateisystem entfernen: rmfs
<i>Mountpunkt</i> |

Anmerkung:

- Diese Prozedur ist erforderlich, wenn Sie einen Datenträger mit einem JFS einlegen.
- Wenn Sie ein JFS entfernen, werden alle Daten, die in diesem Dateisystem und auf dem wieder beschreibbaren optischen Datenträger enthalten sind, gelöscht.

Dateisystem überprüfen:

Es können Inkonsistenzen in Dateisystemen auftreten, wenn das System gestoppt wird, obwohl Dateisysteme angehängt sind, oder wenn eine Platte beschädigt ist. In solchen Fällen ist es wichtig, die Dateisysteme zu überprüfen, bevor sie angehängt werden.

Sie sollten Ihre Dateisysteme auch in den folgenden Fällen überprüfen:

- Nach einer Störung, z. B. wenn ein Benutzer nicht in ein Verzeichnis wechseln kann, das die Berechtigungen (uid) dieses Benutzers hat.
- Vor der Sicherung von Dateisystemen, um Fehler und mögliche Probleme bei der Wiederherstellung zu vermeiden.
- Bei der Installation oder beim Booten des Systems, um sicherzustellen, dass keine Fehler in den Dateien des Betriebssystems vorliegen.

Ein benutzerdefiniertes Dateisystem überprüfen:

Führen Sie die folgenden Schritte aus, um ein benutzerdefiniertes Dateisystem zu überprüfen.

1. Hängen Sie das zu überprüfende benutzerdefinierte Dateisystem ab.
2. Vergewissern Sie sich, dass Sie Schreibzugriff auf die Dateien im Dateisystem haben. Andernfalls kann der Befehl **fsck** beschädigte Dateien nicht reparieren, selbst wenn Sie Systemanfragen zur Reparatur mit Ja beantworten.
3. Verwenden Sie den Direktaufruf **smit fsck**, um auf das Menü **Dateisystem prüfen** zuzugreifen.
4. Führen Sie eine der folgenden Aktionen aus:
 - Geben Sie im Feld **Name des Dateisystems** den Namen eines einzelnen Dateisystems an, das Sie prüfen möchten.
 - Wählen Sie im Feld **Typ des Dateisystems** einen allgemeinen Dateisystemtyp aus, der geprüft werden soll, z. B. Journaled File System (JFS).
5. Wenn Sie Ihre Überprüfung auf die wahrscheinlichsten Kandidaten begrenzen möchten, geben Sie im Feld **Schnelle Überprüfung?** den Wert Ja ein. Bei der schnellen Überprüfung werden nur die Dateisysteme überprüft, die wahrscheinlich Inkonsistenzen enthalten, z. B. die Dateisysteme, die angehängt waren, als das System gestoppt wurde.
6. Geben Sie im Feld **Arbeitsdatei** den Namen einer temporären Datei im Dateisystem an, die nicht geprüft wird.
7. Starten Sie die Überprüfung des Dateisystems.

Stammdateisystem und Dateisystem /usr überprüfen:

Wenn Sie den Befehl **fsck** für das Dateisystem / oder /usr ausführen möchten, müssen Sie das System herunterfahren und anschließend von einem austauschbaren Datenträger erneut booten, da die Dateisysteme / (Stammdateisystem) und /usr nicht von einem aktiven System abgehängt werden können.

Die folgende Prozedur beschreibt, wie der Befehl **fsck** für die Dateisysteme / und /usr in der Wartungshell ausgeführt werden.

1. Fahren Sie Ihr System herunter. (Es ist Rootzugriff erforderlich.)
2. Booten Sie von Ihrem Installationsdatenträger.
3. Wählen Sie im Eingangsmenü die Option **Wartung** aus.
4. Wählen Sie im Menü **Wartung** die Option für den Zugriff auf eine Datenträgergruppe aus.
5. Wählen Sie die Datenträgergruppe rootvg aus. Es wird eine Liste mit logischen Datenträgern angezeigt, die zur ausgewählten Datenträgergruppe gehören.
6. Wählen Sie 2 aus, um auf die Datenträgergruppe zuzugreifen und eine Shell zu starten, bevor Sie Dateisysteme anhängen. In den folgenden Schritten führen Sie den Befehl **fsck** mit den entsprechenden Optionen und Einheiten für das Dateisystem aus. Der Befehl **fsck** überprüft die Konsistenz des Dateisystems und repariert das Dateisystem im Dialogbetrieb. Die Einheit für das Stammdateisystem (/) ist /dev/hd4, und die Einheit für das Dateisystem /usr ist /dev/hd2.
7. Geben Sie Folgendes ein, um das Stammdateisystem (/) zu überprüfen:

```
$ fsck -y /dev/hd4
```

Weniger erfahrene Benutzer sollten das Flag **-y** verwenden (siehe Befehl **fsck**).
8. Geben Sie Folgendes ein, um das Dateisystem /usr zu überprüfen:

```
$ fsck -y /dev/hd2
```
9. Wenn Sie andere Dateisysteme in rootvg überprüfen möchten, geben Sie den Befehl **fsck** mit den entsprechenden Einheiten ein. Die Einheit für /tmp ist /dev/hd3, und die Einheit für /var ist /dev/hd9var.
10. Führen Sie einen Warmstart durch, wenn die Überprüfung der Dateisysteme abgeschlossen ist.

Größe eines Dateisystems in der Stammdatenträgergruppe verringern:

Sie können alle *Dateisysteme* ganz einfach auf ihre Mindestgröße verkleinern, indem Sie beim Wiederherstellen des Basisbetriebssystems von der Sicherung die Option **Verkleinern** auf **Ja** setzen.

Sie können alle *Dateisysteme* ganz einfach auf ihre Mindestgröße verkleinern, indem Sie beim Wiederherstellen des Basisbetriebssystems von der Sicherung die Option **Verkleinern** auf **Ja** setzen. Die Option **Verkleinern** und die in diesem Szenario beschriebene Prozedur schließen sich gegenseitig aus. Wenn Sie die Option **Verkleinern** nach Ausführen der folgenden Prozedur auf **Ja** setzen, überschreibt die Installation Ihre Änderungen in der Datei `/image.data`.

Im Folgenden wird beschrieben, wie Sie die Größe eines ausgewählten `rootvg`-Dateisystems manuell verringern. Sie werden ein Dateisystem ermitteln, das nicht den gesamten Speicherbereich nutzt, der ihm zugeordnet ist. Anschließend werden Sie basierend auf dem Speicherbereich, der in dem Dateisystem tatsächlich genutzt wird, Speicherbereich für die Stammdatenträgergruppe freigeben. Im Rahmen dieser Prozedur müssen Sie die Datenträgergruppen sichern und das Betriebssystem mit den geänderten Zuordnungen erneut installieren.

Achtung: Diese Prozedur erfordert das Herunterfahren und erneute Installieren des Basisbetriebssystems. Um sich vor einem Daten- oder Funktionalitätsverlust zu schützen, sollten Sie für die erneute Installation des Betriebssystems einen Zeitpunkt wählen, der so wenig wie möglich Auswirkungen auf Ihren Arbeitsablauf hat. Bevor Sie das Betriebssystem erneut installieren, müssen Sie sicherstellen, dass Sie zuverlässige Sicherungen Ihrer Daten und aller angepassten Anwendungen und Datenträgergruppen besitzen.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Erstellen Sie von jedem Dateisystem, das *nicht* in der Stammdatenträgergruppe (`rootvg`) enthalten ist, eine gesonderte Sicherung. Die gesonderten Sicherungen gewährleisten die Integrität Ihrer Dateisysteme.
2. Überprüfen Sie mit dem folgenden Befehl, welche Dateisysteme in der Stammdatenträgergruppe nicht den gesamten Speicherbereich nutzen, der ihnen zugeordnet ist. Sie müssen hierfür Rootberechtigung besitzen:

```
df -k
```

Das Flag **-k** zeigt die Größe der Dateisysteme in KB an. Das Ergebnis könnte wie folgt aussehen:

| Filesystem | 1024-blocks | Free | %Used | Iused | %Iused | Mounted on |
|--------------|-------------|--------|-------|-------|--------|------------|
| /dev/hd4 | 196608 | 4976 | 98% | 1944 | 2% | / |
| /dev/hd2 | 1769472 | 623988 | 65% | 36984 | 9% | /usr |
| /dev/hd9var | 163840 | 65116 | 61% | 676 | 2% | /var |
| /dev/hd3 | 65536 | 63024 | 4% | 115 | 1% | /tmp |
| /dev/hd1 | 49152 | 8536 | 83% | 832 | 7% | /home |
| /proc | - | - | - | - | - | /proc |
| /dev/hd10opt | 32768 | 26340 | 20% | 293 | 4% | /opt |

Wenn Sie sich die Ergebnisse genau ansehen, erkennen Sie, dass für das Dateisystem, das an `/usr` angehängt ist, eine große Anzahl freier Blöcke und nur ein relativ geringer Prozentsatz als genutzter Speicherbereich angezeigt wird. Sie entschließen sich, eine beträchtliche Anzahl von Blöcken freizugeben, indem Sie die Anzahl der Partitionen verringern, die dem Dateisystem `/usr` zugeordnet sind.

3. Überprüfen Sie den Inhalt der Datei `/etc/filesystems`, um sicherzustellen, dass alle Dateisysteme in der Stammdatenträgergruppe angehängt sind. Falls nicht alle Dateisysteme angehängt sind, sind diese nicht in dem erneut installierten System enthalten.
4. Erstellen Sie eine Datei `/image.data`, in der Sie alle aktiven Dateisysteme in der Stammdatenträgergruppe auflisten, die in die Installationsprozedur aufgenommen werden sollen. Geben Sie den folgenden Befehl ein:

```
mkszfile
```


- Öffnen Sie die Datei /image.data in Ihrem Editor.
- Suchen Sie nach der Zeichenfolge usr, um die Zeilengruppe "lv_data" zu finden, die zum Dateisystem /usr gehört. Verwenden Sie die Zahlen aus dieser Zeilengruppe als Grundlage, um festzustellen, um welche Anzahl Sie die logischen Partitionen des Dateisystems /usr verringern können. Die Standardgröße jeder zusätzlichen logischen Partition ist in dem Eintrag PP_SIZE der Datei /image.data definiert. Ihre Datei /image.data könnte wie folgt aussehen:

```
lv_data:
VOLUME_GROUP= rootvg
LV_SOURCE_DISK_LIST= hdisk0
LV_IDENTIFIER= 00042345d300bf15.5
LOGICAL_VOLUME= hd2
VG_STAT= active/complete
TYPE= jfs
MAX_LPS= 32512
COPIES= 1
LPS= 108
STALE_PPs= 0
INTER_POLICY= minimum
INTRA_POLICY= center
MOUNT_POINT= /usr
MIRROR_WRITE_CONSISTENCY= on/ACTIVE
LV_SEPARATE_PV= yes
PERMISSION= read/write
LV_STATE= opened/syncd
WRITE_VERIFY= off
PP_SIZE= 16
SCHED_POLICY= parallel
PP= 108
BB_POLICY= relocatable
RELOCATABLE= yes
UPPER_BOUND= 32
LABEL= /usr
MAPFILE=
LV_MIN_LPS= 70
STRIPE_WIDTH=
STRIP_SIZE=
```

Diesem logischen Datenträger sind 108 logische Partitionen zugeordnet (LPS=108).

- Ermitteln Sie die Anzahl benötigter logischer Partitionen anhand der im Dateisystem /usr vorhandenen Daten. Verwenden Sie dazu das Ergebnis aus Schritt 2. Mit dem folgenden Befehl können Sie die vorhandenen Dateigrößen speziell für das Dateisystem /usr anzeigen:

```
df -k /usr
```

Das Ergebnis enthält dieselben Zahlen (in KB) wie das Ergebnis für Dateisystem /usr aus Schritt 2, z. B.:

| Filesystem | 1024-blocks | Free | %Used | Iused | %Iused | Mounted on |
|------------|-------------|--------|-------|-------|--------|------------|
| /dev/hd2 | 1769472 | 623988 | 65% | 36984 | 9% | /usr |

- Ziehen Sie den freien Speicherbereich von der Gesamtanzahl der zugeordneten 1024-Blöcke ab:

$$1769472 - 623988 = 1145484$$

- Addieren Sie den Wert, den Sie in Zukunft schätzungsweise für weitere Daten in diesem Dateisystem benötigen. Addieren Sie für dieses Beispiel 200000 zu dem Ergebnis.

$$1145484 + 200000 = 1345484$$

- Dividieren Sie das Ergebnis durch die Größe der logischen Partitionen in Bytes (16*1024), um die Mindestanzahl erforderlicher logischer Partitionen zu berechnen:

$$1345484 / 16384 = 82,121826171875$$

Verwenden Sie dieses Ergebnis (aufgerundet), um die Anzahl benötigter logischer Partitionen erneut zu definieren (LPS=83).

- Ändern Sie in Ihrer Datei image.data das Feld LPS von 108 in 83.

9. Suchen Sie die Dateigruppe "fs_data" für das Dateisystem /usr. Ihre Zeilengruppe "fs_data" könnte wie folgt aussehen:

```
fs_data:
FS_NAME= /usr
FS_SIZE= 3538944
FS_MIN_SIZE= 2290968
FS_LV= /dev/hd2
FS_FS= 4096
FS_NBPI= 4096
FS_COMPRESS= no
FS_BF= false
FS_AGSIZE= 8
```

10. Berechnen Sie die Größe des Dateisystems (FS_SIZE), indem Sie die Größe der physischen Partition (PP_SIZE) mit 2 (der Anzahl der 512-Byte-Blöcke, die von den physischen Partitionen verwendet werden) und dann mit der Anzahl der logischen Partitionen (LPs) multiplizieren. Gemäß den in diesem Beispiel verwendeten Werten ist die Berechnung wie folgt:

```
PP_SIZE * 512-Blöcke * LPs = FS_SIZE
16384 * 2 * 83 = 2719744
```

11. Ändern Sie in Ihrer Datei image.data das Feld FS von 3538944 in 2719744.

12. Berechnen Sie die Mindestgröße des Dateisystems (FS_MIN_SIZE) ausgehend von der tatsächlichen Größe der aktuellen Daten, die im Dateisystem /usr verwendet werden, wie folgt:

- a. Berechnen Sie die Mindestanzahl benötigter Partitionen. Gemäß den in diesem Beispiel verwendeten Werten ist die Berechnung wie folgt:

```
belegte_Größe (siehe Schritt 7a) / PP_SIZE = Partitionen
1145484 / 16384 = 69,914794921875
```

- b. Berechnen Sie die erforderliche Mindestgröße für diese Anzahl von Partitionen. Wenn Sie das Ergebnis der vorherigen Berechnung auf 70 aufrunden, ergibt sich folgende Formel:

```
PP_SIZE * 512-Blöcke * Partitionen = FS_MIN_SIZE
16384 * 2 * 70 = 2293760
```

13. Ändern Sie in Ihrer Datei image.data das Feld FS_MIN_SIZE von 2290968 in 2293760.

14. Speichern Sie Ihre Änderungen und verlassen Sie den Editor.

15. Hängen Sie alle Dateisysteme ab, die nicht in der Stammdatenträgergruppe enthalten sind.

16. Wenn Sie benutzerdefinierte Datenträgergruppen besitzen, geben Sie die folgenden Befehle ein, um diese Datenträgergruppen abzuhängen und zu exportieren:

```
varyoffvg Name_der_Datenträgergruppe
exportvg Name_der_Datenträgergruppe
```

17. Legen Sie ein Band in das Bandlaufwerk ein, und geben Sie dann den folgenden Befehl ein, um eine vollständige Systemdatensicherung einzuleiten:

```
mksysb /dev/rmt0
```

Bei dieser Art der Sicherung wird die Dateisystemgröße aufgezeichnet, die Sie in der Datei /image.data festgelegt haben und die später für die erneute Installation des Systems verwendet wird.

Anmerkung: Zum Einleiten der Sicherung müssen Sie den Befehl **mksysb** in der Befehlszeile ausführen. Wenn Sie ein Systemverwaltungstool wie SMIT verwenden, erstellt die Sicherung eine neue Datei image.data und überschreibt damit alle vorgenommenen Änderungen.

18. Verwenden Sie diese Sicherung, um das Betriebssystem mit der Option **Mit aktuellen Systemeinstellungen installieren** zu installieren. Überprüfen Sie während der Installation, ob die folgenden Optionen ordnungsgemäß definiert sind:

- **Zuordnungen verwenden** muss auf **Nein** gesetzt sein.
- **Dateisysteme verkleinern** muss auf **Nein** gesetzt sein.

Nähere Informationen zur Installationsprozedur finden Sie im Abschnitt "Installing system backups" in der Veröffentlichung "Installation and Migration".

19. Nach der Installation des Betriebssystems müssen Sie das System im normalen Modus erneut booten. Damit ist das Dateisystem /usr zwar verkleinert, aber Ihre benutzerdefinierten Dateisysteme sind noch nicht verfügbar.
20. Hängen Sie alle Dateisysteme mit dem folgenden Befehl an:

```
mount all
```

Wenn für Dateisysteme, die bereits angehängt sind, die Nachricht Einheit aktiv angezeigt wird, können Sie diese ignorieren.

Damit ist das Dateisystem /usr verkleinert, der Stammdatenträgergruppe steht mehr freier Speicherbereich zur Verfügung, und Ihre Dateisysteme sind verwendbar.

Zugehörige Konzepte:

„Speicherung logischer Datenträger“ auf Seite 409

Logische Datenträger sind Gruppen von Informationen auf physischen Datenträgern.

Zugehörige Informationen:

Creating a Root Volume Group Backup to Tape or File

/image.data file description

mkszfile command

mksysb command

Fehlerbehebung bei Dateisystemen

Verwenden Sie die folgenden Methoden für die Behebung einiger der grundlegenden Probleme, die bei Ihren Dateisystemen auftreten können. Wenn Ihr Problem in den verfügbaren Fehlerbehebungsinformationen nicht adressiert wird, wenden Sie sich an Ihren IBM Ansprechpartner.

Überlauf eines benutzerdefinierten Dateisystems korrigieren:

Verwenden Sie diese Prozedur, um ein überlaufendes benutzerdefiniertes Dateisystem zu korrigieren.

1. Entfernen Sie alte Sicherungsdateien und Kerndateien. Das folgende Beispiel entfernt alle Dateien vom Typ *.bak, *.bak, a.out, core, * und ed.hup.

```
find / \( -name "*.bak" -o -name core -o -name a.out -o \  
-name "...*" -o -name "*.bak" -o -name ed.hup \) \  
-atime +1 -mtime +1 -type f -print | xargs -e rm -f
```

2. Führen Sie den Befehl **skulker** im Rahmen des **cron**-Prozesses aus, und entfernen Sie Dateien, die nicht mehr benötigt werden oder temporäre Dateien sind, um zu verhindern, dass es regelmäßig zu einem Überlauf der Platte durch Dateien kommt.

Der Befehl **skulker** löscht Dateien im Verzeichnis /tmp, Dateien, die ein angegebenes Alter überschreiten, Dateien vom Typ a.out, Kerndateien und Dateien vom Typ ed.hup. Der Befehl wird täglich im Rahmen einer vom Befehl **cron** ausgeführten Abrechnungsprozedur zu Zeiten mit geringer Systemauslastung ausgeführt (sofern Sie die Abrechnung aktiviert haben).

Der Dämon **cron** führt an bestimmten Tagen zu bestimmten Zeiten Shellbefehle aus. Regelmäßig geplante Befehle wie **skulker** können gemäß Anweisungen in den Dateien crontab angegeben werden. Übergeben Sie crontab-Dateien mit dem Befehl **crontab**. Zum Editieren einer crontab-Datei müssen Sie Rootberechtigung besitzen.

Zugehörige Tasks:

„Abrechnungssystem konfigurieren“ auf Seite 176

Sie können ein Abrechnungssystem konfigurieren.

Ein beschädigtes Dateisystem korrigieren:

Dateisysteme können beschädigt werden, wenn die I-Node- oder Superblockinformationen für die Verzeichnisstruktur des Dateisystems beschädigt werden.

Diese Beschädigung kann durch einen hardwarebezogenen Fehler oder ein beschädigtes Programm verursacht werden, das auf die I-Node- oder Superblockinformationen direkt zugreift. (Programme, die in Assembler oder C geschrieben sind, können das Betriebssystem umgehen und direkt auf die Hardware schreiben.) Wenn das System Daten in einem bestimmten Dateisystem nicht findet, nicht lesen oder schreiben kann, deutet dies auf ein beschädigtes Dateisystem hin.

Zum Korrigieren eines beschädigten Dateisystems müssen Sie das Problem diagnostizieren und anschließend beheben. Der Befehl **fsck** führt systemnahe Diagnose- und Reparaturfunktionen aus.

Im Folgenden wird die Prozedur zum Korrigieren eines beschädigten Dateisystems beschrieben:

1. Melden Sie sich als Root an, und hängen Sie das beschädigte Dateisystem mit einem der folgenden SMIT-Direktaufrufe ab: **smit unmountfs** (für ein Dateisystem auf einem Festplattenlaufwerk) oder **smit unmountdsk** (für ein Dateisystem auf einem austauschbaren Datenträger).
2. Ermitteln Sie den Dateisystemdefekt mit dem Befehl **fsck**. Im folgenden Beispiel überprüft der Befehl **fsck** das abgehängte Dateisystem auf der Einheit `/dev/myfilelv`:

```
fsck /dev/myfilelv
```

Der Befehl **fsck** überprüft und repariert inkonsistente Dateisysteme im Dialogbetrieb. Das Dateisystem ist normalerweise konsistent, und der Befehl **fsck** zeigt nur die Anzahl der Dateien sowie die benutzten und freien Blöcke im Dateisystem an. Ist das Dateisystem inkonsistent, zeigt der Befehl **fsck** Informationen über die festgestellten Inkonsistenzen an und fordert Sie auf, die Berechtigung zur Korrektur zu erteilen. Bei der Ausführung des Befehls **fsck** werden Korrekturen äußerst vorsichtig vorgenommen und Aktionen vermieden, die den Verlust gültiger Daten nach sich ziehen könnten. In bestimmten Fällen empfiehlt der Befehl **fsck** jedoch, die beschädigte Datei zu löschen. Eine Liste der Inkonsistenzen, nach denen dieser Befehl sucht, finden Sie in der Beschreibung des Befehls **fsck** in der Veröffentlichung *Commands Reference, Volume 2*.

3. Wenn das Dateisystem nicht repariert werden kann, stellen Sie es aus einer Sicherung wieder her.
Achtung: Wenn Sie ein Dateisystem aus einer Sicherung wiederherstellen, werden alle Dateien, die auf der Platte gespeichert sind, gelöscht und ersetzt.

Verwenden Sie zum Wiederherstellen des Dateisystems aus einer Sicherung den SMIT-Direktauftrag **smit restfilesys** oder die im folgenden Beispiel gezeigte Folge von Befehlen:

```
mkfs /dev/myfilelv  
mount /dev/myfilelv /myfilesys  
cd /myfilesys  
restore -r
```

In diesem Beispiel erstellt der Befehl **mkfs** ein neues Dateisystem auf der Einheit `/dev/myfilelv` und initialisiert den Datenträgerkennsatz, den Dateisystemkennsatz und den Startblock. Der Befehl **mount** hängt `/dev/myfilelv` am Mountpunkt **myfilesys** an, und der Befehl **restore** extrahiert das Dateisystem aus der Sicherung.

Wenn die Sicherung mithilfe von inkrementellen Dateisystemsicherungen erstellt wurde, müssen Sie die Sicherungen in aufsteigender Reihenfolge (z. B. Sicherung 0, Sicherung 1, Sicherung 2) wiederherstellen. Wenn Sie **smit restfilesys** zum Wiederherstellen eines vollständigen Dateisystems verwenden, geben Sie das Zielverzeichnis, die Wiederherstellungseinheit (eine andere Einheit als `/dev/rfd0`) und die Anzahl der Blöcke an, die in einer einzelnen Eingabeoperation gelesen werden sollen.

Zugehörige Tasks:

„Benutzerdateien aus einem Sicherungsbild wiederherstellen“ auf Seite 33

Wenn Sie ein versehentlich gelöscht Sicherungsbild wiederherstellen müssen, ist die größte Schwierigkeit festzustellen, welches der Sicherungsbänder diese Datei enthält. Mit dem Befehl **restore -T** können Sie den Inhalt eines Archivs auflisten. Es empfiehlt sich, die Datei im Verzeichnis `/tmp` wiederherzustellen.

len, damit Sie die anderen Dateien des Benutzers nicht versehentlich überschreiben.

Einen beschädigten Dateitypanzeiger im Superblock des Dateisystems korrigieren:

Falls der Superblock eines Dateisystems beschädigt ist, kann nicht auf das Dateisystem zugegriffen werden. Ein beschädigter Dateitypanzeiger (Magic Number) im Superblock des Dateisystems kann korrigiert werden.

Die meisten Fehler im Superblock können nicht behoben werden. Die folgende Prozedur beschreibt, wie Sie einen Superblock in einem JFS-Dateisystem reparieren, wenn das Problem durch einen beschädigten Dateitypanzeiger verursacht wird. Falls der primäre Superblock eines JFS2-Dateisystems beschädigt ist, können Sie mit dem Befehl **fsck** den sekundären Superblock automatisch kopieren und den primären Superblock reparieren.

Im folgenden Szenario wird davon ausgegangen, dass `/home/myfs` ein Dateisystem vom Typ JFS auf dem physischen Datenträger `/dev/lv02` ist.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

1. Hängen Sie das Dateisystem `/home/myfs`, von dem Sie vermuten, das es beschädigt ist, mit dem folgenden Befehl ab:
`umount /home/myfs`
2. Um sich zu vergewissern, dass das Dateisystem wirklich beschädigt ist, führen Sie den Befehl **fsck** für das Dateisystem aus. Beispiel:
`fsck -p /dev/lv02`

Falls das Problem durch einen beschädigten Superblock hervorgerufen wird, gibt der Befehl **fsck** eine der folgenden Nachrichten zurück:

```
fsck: Kein AIXV5-Dateisystem
```

ODER

```
Kein bekannter Dateisystemtyp
```

3. Führen Sie, wie im folgenden Beispiel gezeigt, den Befehl **od** aus, um den Superblock des Dateisystems anzuzeigen. Sie müssen hierfür Rootberechtigung besitzen:
`od -x -N 64 /dev/lv02 +0x1000`

Das Flag `-x` zeigt die Ausgabe im Hexadezimalformat an, und das Flag `-N` weist das System an, nicht mehr als 64 eingegebene Bytes ab dem Offset-Parameter (+) zu formatieren, der die Stelle in der Datei angibt, mit der die Dateiausgabe beginnt. Im Folgenden sehen Sie eine Beispielausgabe:

```
0001000 1234 0234 0000 0000 0000 4000 0000 000a
0001010 0001 8000 1000 0000 2f6c 7633 0000 6c76
0001020 3300 0000 000a 0003 0100 0000 2f28 0383
0001030 0000 0001 0000 0200 0000 2000 0000 0000
0001040
```

In dieser Ausgabe finden Sie den beschädigten Dateitypanzeiger an Position `0x1000` (`1234 0234`). Falls beim Erstellen des Dateisystems alle Standardwerte akzeptiert wurden, muss der Dateitypanzeiger `0x43218765` sein. Sollten Sie Standardwerte überschrieben haben, muss der Dateitypanzeiger `0x65872143` sein.

4. Überprüfen Sie mit dem Befehl **od**, ob der sekundäre Superblock einen gültigen Dateitypanzeiger enthält. Im Folgenden sehen Sie einen Beispielbefehl und die zugehörige Ausgabe:

```
$ od -x -N 64 /dev/lv02 +0x1f000
001f000 6587 2143 0000 0000 0000 4000 0000 000a
001f010 0001 8000 1000 0000 2f6c 7633 0000 6c76
001f020 3300 0000 000a 0003 0100 0000 2f28 0383
001f030 0000 0001 0000 0200 0000 2000 0000 0000
001f040
```

Diese Ausgabe enthält an Position 0x1f000 den korrekten Dateitypanzeiger.

5. Kopieren Sie den sekundären Superblock in den primären Superblock. Im Folgenden sehen Sie einen Beispielbefehl und die zugehörige Ausgabe:

```
$ dd count=1 bs=4k skip=31 seek=1 if=/dev/lv02 of=/dev/lv02
dd: 1+0 records in.
dd: 1+0 records out.
```

6. Verwenden Sie den Befehl **fsck**, um inkonsistente Dateien unter Verwendung des sekundären Superblocks zu bereinigen. Beispiel:

```
fsck /dev/lv02 2>&1 | tee /tmp/fsck.errs
```

Zugehörige Informationen:

fsck command

od command

Plattenüberlauf

Ein Plattenüberlauf tritt auf, wenn zu viele Dateien den reservierten Speicherplatz füllen. Der Grund hierfür kann ein "Ausreißerprozess" sein, der viele nicht benötigte Dateien erstellt.

Sie können das Problem mit den folgenden Prozeduren beheben:

Anmerkung: Zum Entfernen von Prozessen, die nicht Ihre eigenen sind, benötigen Sie Rootberechtigung.

Zugehörige Konzepte:

„Befehl für die automatische Bereinigung von Dateisystemen“ auf Seite 397

Mit dem Befehl **skulker** können Sie Dateisysteme bereinigen und die nicht erwünschten Dateien entfernen.

Problemprozesse ermitteln:

Verwenden Sie die folgende Prozedur, um Problemprozesse einzugrenzen.

1. Geben Sie Folgendes ein, um den Prozessstatus zu überprüfen und die Prozesse zu ermitteln, die die Verursacher des Problems sein könnten:

```
ps -ef | pg
```

Der Befehl **ps** zeigt den Prozessstatus an. Das Flag **-e** schreibt Informationen zu allen Prozessen (mit Ausnahme von Kernelprozessen) an, und das Flag **-f** generiert eine vollständige Liste der Prozesse mit dem Befehlsnamen und den Parametern, die beim Erstellen des Prozesses verwendet wurden. Der Befehl **pg** begrenzt die Ausgabe auf jeweils eine Seite, so dass die Informationen nicht zu schnell aus dem Sichtbereich verschoben werden.

Suchen Sie nach System- oder Benutzerprozessen, die übermäßig viel Systemressourcen, z. B. CPU-Zeit, beanspruchen. Systemprozesse wie **sendmail**, **routed** und **lpd** scheinen die Systemprozesse zu sein, die am häufigsten unter den "Ausreißern" zu finden sind.

2. Geben Sie Folgendes ein, um nach Benutzerprozessen zu suchen, die mehr CPU als erwartet beanspruchen:

```
ps -u
```

3. Notieren Sie die Prozess-IDs (PID) der Problemprozesse.

Einen Prozess beenden:

Sie können Problemprozesse beenden.

Verwenden Sie die folgende Prozedur, um einen Problemprozess zu beenden:

1. Geben Sie Folgendes ein, um den Prozess zu beenden, der das Problem verursacht:

```
kill -9 PID
```

PID steht für die ID des Problemprozesses.

2. Geben Sie Folgendes ein, um die Dateien zu entfernen, die der Prozess erstellt hat:

```
rm Datei1 Datei2 Datei3
```

Datei1 Datei2 Datei3 steht für die Namen von prozessbezogenen Dateien.

Freigabe von Speicherplatz ohne Beendigung eines Prozesses:

Wenn Sie die Blöcke zurückfordern möchten, die einer aktiven Datei zugeordnet sind, ohne den Prozess zu beenden, müssen Sie die Ausgabe eines anderen Befehls in die Datei umleiten. Die Datenumleitung schneidet die Datei ab und gibt die belegten Speicherblöcke frei.

Wenn die aktive Datei aus dem Dateisystem entfernt wird, bleiben die der Datei zugeordneten Blöcke so lange zugeordnet, bis die letzte offene Referenz entfernt wird, indem der Prozess die Datei schließt oder der Prozess, der die Datei geöffnet hat, beendet wird. Wenn ein "Ausreißerprozess" in eine Datei schreibt und die Datei entfernt wird, werden die der Datei zugeordneten Blöcke erst dann freigegeben, wenn der Prozess beendet wird.

Beispiel:

```
$ ls -l
total 1248
-rwxrwxr-x    1 web  staff   1274770 Jul 20 11:19 datafile
$ date > datafile
$ ls -l
total 4
-rwxrwxr-x    1 web  staff      29 Jul 20 11:20 datafile
```

Die Ausgabe des Befehls **date** ersetzt den früheren Inhalt der Datei *datafile*. Die für die abgeschnittene Datei zurückgemeldeten Blöcke spiegeln einen Größenunterschied von 1248> zu 4 wider. Wenn der Ausreißerprozess weiterhin Informationen an diese neu abgeschnittene Datei anfügt, erzeugt der nächste Befehl **ls** die folgenden Ergebnisse:

```
$ ls -l
total 8
-rwxrwxr-x    1 web  staff   1278866 Jul 20 11:21 datafile
```

Die Größe der Datei *datafile* gibt an, wie viel Daten der Ausreißerprozess angefügt hat, aber die Anzahl der zugeordneten Blöcke ist gering. Die Datei *datafile* enthält jetzt eine Lücke. Dateilücken sind Bereiche der Datei, denen keine Plattenblöcke zugeordnet sind.

Überlauf des Stammdateisystems (/):

Überprüfen Sie Folgendes, wenn die Speicherkapazität des Stammdateisystems (/) erschöpft ist.

- Verwenden Sie den folgenden Befehl, um den Inhalt der Datei */etc/security/failedlogin* zu lesen:

```
who /etc/security/failedlogin
```

Der Umstand, dass nicht grafikfähige Terminals zu schnell Prozesse erzeugen, kann dazu führen, dass Einträge zu gescheiterten Anmeldungen erzeugt werden. Führen Sie den folgenden Befehl aus, um die Datei zu löschen, nachdem Sie die Ausgabe gelesen oder gespeichert haben:

```
cp /dev/null /etc/security/failedlogin
```

- Suchen Sie im Verzeichnis /dev nach einem falsch geschriebenen Einheitennamen. Wenn ein Einheitenname falsch eingegeben wird, z. B. rmt0 anstelle von rmt0, wird eine Datei mit dem Namen rmt0 in /dev erstellt. Der Befehl wird normalerweise so lange ausgeführt, bis das gesamte Stammdateisystem gefüllt ist. Erst dann schlägt er fehl. /dev gehört zum Stammdateisystem (/). Suchen Sie nach Einträgen, die keine Einheiten bezeichnen (keine Primärzahl (major number) oder Sekundärzahl (minor number)) haben. Verwenden Sie den folgenden Befehl, um das System auf eine solche Situation zu überprüfen:

```
cd /dev  
ls -l | pg
```

An derselben Position, an der eine Dateigröße für eine gewöhnliche Datei angegeben wird, hat eine Einheitsdatei zwei Zahlen, die durch ein Komma getrennt sind. Beispiel:

```
crw-rw-rw- 1 root system 12,0 Oct 25 10:19 rmt0
```

Wenn der Dateiname oder die Position der Größenangabe auf eine ungültige Einheit hinweist, wie im folgenden Beispiel gezeigt, entfernen Sie die zugehörige Datei:

```
crw-rw-rw- 1 root system 9375473 Oct 25 10:19 rmt0
```

Anmerkung:

- Entfernen Sie keine gültigen Einheitennamen aus dem Verzeichnis /dev. Ein Indikator für eine ungültige Einheit ist eine zugeordnete Dateigröße, die den Wert von 500 Bytes überschreitet.
- Wenn die Systemprotokollierung aktiv ist, kann sich das Standardverzeichnis /audit schnell füllen und der Aufmerksamkeit bedürfen.
- Suchen Sie mit dem Befehl **find** nach sehr großen Dateien, die entfernt werden können. Verwenden Sie beispielsweise den folgenden Befehl, um alle Dateien im Stammverzeichnis (/) zu suchen, die größer sind als 1 MB:

```
find / -xdev -size +2048 -ls |sort -r -n +6
```

Dieser Befehl sucht alle Dateien, die größer sind als 1 MB, und sortiert sie in umgekehrter Reihenfolge, angefangen bei der größten Datei. Andere Flags für den Befehl find, wie z. B. **-newer**, können bei dieser Suche hilfreich sein. Ausführliche Informationen finden Sie in der Beschreibung des Befehls **find** in der Veröffentlichung "Commands Reference".

Anmerkung: Wenn Sie das Stammverzeichnis überprüfen, werden Sie feststellen, dass die Primär- und Sekundärzahlen für Einheiten im Verzeichnis /dev mit echten Dateien und Dateigrößen durchsetzt sind. Primär- und Sekundärzahlen, die durch ein Komma getrennt sind, können ignoriert werden.

Bevor Sie Dateien entfernen, sollten Sie sich mit dem folgenden Befehl vergewissern, dass eine Datei momentan nicht von einem Benutzerprozess verwendet wird.

```
fuser Dateiname
```

Dateiname steht für den Namen der verdächtigen großen Datei. Wenn Sie versuchen, eine Datei zu entfernen, und diese Datei zu diesem Zeitpunkt geöffnet ist, wird sie nur aus der Verzeichnisliste entfernt. Die Blöcke, die dieser Datei zugeordnet sind, werden erst freigegeben, wenn der Prozess, der die Datei offen hält, beendet wird.

Überläufe im Dateisystem /var beheben:

Überprüfen Sie Folgendes, wenn die Speicherkapazität des Dateisystems (/var) erschöpft ist.

- Mit dem Befehl find können Sie große Dateien im Verzeichnis /var suchen. Beispiel:

```
find /var -xdev -size +2048 -ls | sort -r +6
```

Ausführliche Informationen finden Sie in der Beschreibung des Befehls **find** in der Veröffentlichung "Commands Reference".

- Suchen Sie nach veralteten oder verwaisten Dateien in /var/tmp.
- Überprüfen Sie die Größe der Datei /var/adm/wtmp, in der alle Anmeldungen, fernanmeldungen und Telnet-Sitzungen protokolliert werden. Das Protokoll wächst auf unbestimmte Größe an, sofern die

Systemabrechnung nicht aktiv ist. Wenn die Systemabrechnung aktiv ist, wird der Inhalt des Protokolls jede Nacht gelöscht. Der Inhalt der Datei `/var/adm/wtmp` kann gelöscht oder bearbeitet werden, um alte und nicht erwünschte Informationen zu entfernen. Verwenden Sie den folgenden Befehl, um den Inhalt des Protokolls zu löschen:

```
cp /dev/null /var/adm/wtmp
```

Wenn Sie die Datei `/var/adm/wtmp` bearbeiten möchten, müssen Sie die Datei mit dem folgenden Befehl vorübergehend kopieren:

```
/usr/sbin/acct/fwtmp < /var/adm/wtmp >/tmp/out
```

Bearbeiten Sie die Datei `/tmp/out`, um nicht erwünschte Einträge zu entfernen, und ersetzen Sie anschließend die ursprüngliche Datei mit dem folgenden Befehl:

```
/usr/sbin/acct/fwtmp -ic < /tmp/out > /var/adm/wtmp
```

- Löschen Sie den Inhalt des Fehlerprotokolls im Verzeichnis `/var/adm/ras` mit der folgenden Prozedur. Der Inhalt des Fehlerprotokolls wird nur gelöscht, wenn er manuell gelöscht wird.

Anmerkung: Verwenden Sie nicht den Befehl `cp /dev/null`, um den Inhalt des Fehlerprotokolls zu löschen. Eine Datei `errlog` mit der Größe `null` inaktiviert die Fehlerprotokollierungsfunktionen des Betriebssystems und muss durch die Version aus einer Sicherung ersetzt werden.

1. Stoppen Sie den Fehlerdämon mit dem folgenden Befehl:

```
/usr/lib/errstop
```

2. Entfernen Sie die Fehlerprotokolldatei oder verschieben Sie sie in ein anderes Dateisystem. Verwenden Sie dazu die folgenden Befehle:

```
rm /var/adm/ras/errlog
```

oder

```
mv /var/adm/ras/errlog Dateiname
```

Dateiname steht für den Namen der verschobenen Datei `errlog`.

Anmerkung: Die Langzeitfehlerdaten werden gelöscht, wenn Sie die Fehlerprotokolldatei entfernen.

3. Starten Sie den Fehlerdämon mit dem folgenden Befehl erneut:

```
/usr/lib/errdemon
```

Anmerkung: Wenn es für Sie in Betracht kommt, können Sie mit den folgenden Einträgen in `cron` das Fehlerprotokoll begrenzen:

```
0 11 * * * /usr/bin/errclear -d S,0 30
```

```
0 12 * * * /usr/bin/errclear -d H 90
```

- Prüfen Sie, ob die Datei `trcfile` in diesem Verzeichnis sehr groß ist. Wenn sie groß ist und derzeit kein Trace durchgeführt wird, können Sie die Datei mit dem folgenden Befehl entfernen:

```
rm /var/adm/ras/trcfile
```

- Wenn `hd6` als Speicherausgangseinheit definiert ist (Standardeinstellung), kann es eine Reihe von Dateien des Typs `vmcore*` im Verzeichnis `/var/adm/ras` geben. Wenn diese Dateien sehr alt sind oder Sie diese Dateien nicht mehr aufbewahren möchten, können Sie sie mit dem Befehl `rm` entfernen.

- Überprüfen Sie das Verzeichnis `/var/spool`, das die Dateien für das Warteschlangensubsystem enthält. Löschen Sie das Warteschlangensubsystem mit den folgenden Befehlen:

```
stopsrc -s qdaemon
```

```
rm /var/spool/lpd/qdir/*
```

```
rm /var/spool/lpd/stat/*
```

```
rm /var/spool/qdaemon/*
```

```
startsrc -s qdaemon
```

- Überprüfen Sie das Verzeichnis `/var/adm/acct`, das die Abrechnungssätze enthält. Wenn die Abrechnung aktiv ist, kann dieses Verzeichnis mehrere große Dateien enthalten.
- Suchen Sie im Verzeichnis `/var/preserve` nach beendeten `vi`-Sitzungen. Im Allgemeinen können diese Dateien unbesorgt entfernt werden. Wenn ein Benutzer eine Sitzung wiederherstellen möchte, können

Sie mit dem Befehl **vi -r** alle wiederherstellbaren Sitzungen auflisten. Zum Wiederherstellen einer bestimmten Sitzung verwenden Sie den Befehl **vi -r** *Dateiname*.

- Ändern Sie die Datei */var/adm/sulog*, in der die Anzahl der versuchten Ausführungen des Befehls **su** und deren Erfolg aufgezeichnet werden. Es handelt sich hierbei um eine unstrukturierte Datei, die Sie mit dem von Ihnen bevorzugten Editor anzeigen und ändern können. Wenn diese Datei entfernt wird, wird sie beim nächsten Versuch, den Befehl **su** auszuführen, erneut erstellt. Ändern Sie die Datei */var/tmp/snmpd.log*, in der Ereignisse während der Ausführung des Dämons **snmpd** aufgezeichnet werden. Wenn diese Datei entfernt wird, wird sie vom Dämon **snmpd** erneut erstellt.

Anmerkung: Die Größe der Datei */var/tmp/snmpd.log* kann begrenzt werden, so dass die Datei nicht unbegrenzt anwächst. Editieren Sie die Datei */etc/snmpd.conf*, um die Anzahl (Bytes) im entsprechenden Abschnitt für die Größe zu ändern.

Zugehörige Konzepte:

„Systemabrechnung“ auf Seite 164

Mit dem Dienstprogramm für Systemabrechnung können Sie Informationen zur Verwendung verschiedener Systemressourcen für Einzelpersonen und Gruppen erfassen und zurückmelden.

Andere Dateisysteme korrigieren und allgemeine Suchtechniken:

Mit dem Befehl **find** und dem Flag **-size** können Sie große Dateien suchen. Wenn in jüngster Vergangenheit ein Dateisystemüberlauf aufgetreten ist, können Sie mit dem Flag **-newer** kürzlich geänderte Dateien suchen.

Verwenden Sie den folgenden Befehl **touch**, um eine Datei zu erzeugen, die das Flag **-newer** als Vergleich für die Suche verwenden kann:

```
touch mmddhhmm Dateiname
```

mm steht für den Monat, *dd* für das Datum, *hh* für die Stunde im 24-Stunden-Format, *mm* für die Minute und *Dateiname* für den Namen der Datei, die Sie mit dem Befehl **touch** erstellen.

Nachdem Sie die Datei mit **touch** erstellt haben, können Sie mit dem folgenden Befehl nach neueren großen Dateien suchen:

```
find /Name_des_Dateisystems -xdev -newer Name_der_touch-Datei -ls
```

Sie können mit dem Befehl **find** auch Dateien suchen, die in den letzten 24 Stunden geändert wurden. Dies zeigt das folgende Beispiel:

```
find /Name_des_Dateisystems -xdev -mtime 0 -ls
```

Anhängen (Mount)

Wenn Dateisysteme, Dateien, Verzeichnisse, Einheiten und Gerätedateien *angehängt* (Mount) werden, können diese an einer bestimmten Position bereitgestellt werden. Es ist die einzige Möglichkeit, um den Zugriff auf ein Dateisystem zu gewähren.

Der Befehl **mount** weist das Betriebssystem an, ein Dateisystem an einem angegebenen Verzeichnis anzuhängen.

Sie können eine Datei oder ein Verzeichnis anhängen, wenn Sie Zugriff auf die anzuhängende Datei bzw. das anzuhängende Verzeichnis und Schreibberechtigung für den Mountpunkt haben. Mitglieder der Gruppe **system** können auch Einheitsmounts (hierbei werden Einheiten oder Dateisysteme über Verzeichnisse angehängt) und die in der Datei */etc/filesystems* beschriebenen Mounts durchführen. Ein Benutzer, der Rootberechtigung besitzt, kann ein Dateisystem anhängen, indem er die Einheit und das Verzeichnis in der Befehlszeile angibt. Die Datei */etc/filesystems* wird verwendet, um Mounts zu definieren, die automatisch während der Systeminitialisierung durchgeführt werden sollen. Mit dem Befehl **mount** können Mounts nach dem Systemstart durchgeführt werden.

Mountpunkte:

Ein *Mountpunkt* ist ein Verzeichnis oder eine Datei, über das bzw. die ein neues Dateisystem, ein Verzeichnis oder eine Datei zugänglich gemacht wird. Wenn Sie ein Dateisystem oder ein Verzeichnis anhängen möchten, muss der Mountpunkt ein Verzeichnis sein. Für das Anhängen einer Datei muss der Mountpunkt eine Datei sein.

Gewöhnlich wird ein Dateisystem, ein Verzeichnis oder eine Datei über einen leeren Mountpunkt angehängt. Dies ist jedoch keine Voraussetzung. Wenn die Datei oder das Verzeichnis, die bzw. das als Mountpunkt dient, Daten enthält, sind diese Daten nicht zugänglich, solange eine andere Datei oder ein anderes Verzeichnis angehängt ist. In der Tat verdeckt die angehängte Datei bzw. das angehängte Verzeichnis die am Mountpunkt enthaltenen Daten. Das ursprüngliche Verzeichnis bzw. die ursprüngliche Datei, das bzw. die als Mountpunkt verwendet wird, ist wieder verfügbar, sobald die angehängte Datei bzw. das angehängte Verzeichnis abgehängt wird.

Wenn ein Dateisystem über ein Verzeichnis angehängt wird, haben die Berechtigungen des Stammverzeichnisses des angehängten Dateisystems eine Vorrangstellung gegenüber den Berechtigungen des Mountpunkts. Die einzige Ausnahme bildet der übergeordnete Verzeichniseintrag .. (Punkt Punkt) im Verzeichnis, das als Mountpunkt verwendet wird. Damit das Betriebssystem auf das neue Dateisystem zugreifen kann, müssen die Informationen zum übergeordneten Verzeichnis des Mountpunkts verfügbar sein.

Wenn das aktuelle Arbeitsverzeichnis beispielsweise /home/frank ist, ändert der Befehl `cd ..` das Arbeitsverzeichnis in /home. Ist das Verzeichnis /home/frank das Stammverzeichnis eines angehängten Dateisystems, muss das Betriebssystem die Informationen zum übergeordneten Verzeichnis im Verzeichnis /home/frank finden, damit der Befehl `cd ..` erfolgreich ist.

Damit Befehle, die Informationen zum übergeordneten Verzeichnis benötigen, erfolgreich ausgeführt werden können, müssen Benutzer im Verzeichnis, das als Mountpunkt dient, Suchberechtigung besitzen. Wenn das Verzeichnis, das als Mountpunkt dient, dem Benutzer die Suchberechtigung verweigert, kann dies unvorhersehbare Ergebnisse haben, insbesondere weil die Berechtigungen des Verzeichnisses, das als Mountpunkt dient, nicht sichtbar sind. Ein häufig auftretender Fehler ist das Scheitern des Befehls `pwd`. Wenn der Benutzer in dem Verzeichnis, das als Mountpunkt dient, keine Suchberechtigung hat, gibt der Befehl `pwd` die folgende Nachricht zurück:

```
pwd: Berechtigung verweigert
```

Dieses Problem kann vermieden werden, indem die Berechtigungen für das Verzeichnis, das als Mountpunkt dient, stets auf 111 gesetzt werden.

Dateisysteme, Verzeichnisse und Dateien anhängen:

Es gibt zwei Typen von Anhängoperationen (oder Mounttypen): Mount auf einem fernen System und lokaler Mount. *Mounts auf einem fernen System* werden auf einem fernen System durchgeführt, an das über eine Telekommunikationsleitung Daten übertragen werden. Für ferne Dateisysteme, z. B. Network File System (NFS), müssen die Dateien exportiert werden, damit sie angehängt werden können. *Lokale Mounts* sind Anhängoperationen (Mounts), die auf dem lokalen System durchgeführt werden.

Jedes Dateisystem ist einer anderen Einheit (logischer Datenträger) zugeordnet. Bevor Sie ein Dateisystem verwenden können, muss es mit der vorhandenen Verzeichnisstruktur (dem Stammdateisystem oder einem anderen, bereits verbundenen Dateisystem) verbunden werden. Die Verbindung wird vom Befehl `mount` hergestellt.

Der Zugriff auf ein Dateisystem, ein Verzeichnis oder eine Datei kann über mehrere Pfade erfolgen. Wenn Sie beispielsweise eine Datenbank haben und mehrere Benutzer diese Datenbank verwenden, kann es hilfreich sein, dieselbe Datenbank mehrfach anzuhängen, d. h. mehrere Mounts zu haben. Jeder Mount muss zu Überwachungszwecken und zur Unterscheidung der Jobs einen eigenen Namen und ein eigenes

Kennwort haben. Dies kann erreicht werden, indem dasselbe Dateisystem über unterschiedliche Mountpunkte angehängt wird. Sie können /home/server/database beispielsweise über die Mountpunkte /home/user1, /home/user2 und /home/user3 anhängen:

```
/home/server/database    /home/user1
/home/server/database    /home/user2
/home/server/database    /home/user3
```

Mithilfe von symbolischen Verbindungen kann ein Dateisystem, ein Verzeichnis oder eine Datei verschiedenen Benutzern bereitgestellt werden. Symbolische Verbindungen werden mit dem Befehl **ln -s** erstellt. Wenn Sie mehrere Benutzer mit einer zentralen Datei verbinden, wird damit sichergestellt, dass alle an der Datei vorgenommenen Änderungen verfügbar sind, wenn ein Benutzer auf die Datei zugreift.

Automatische Mountsteuerung:

Mounts können so konfiguriert werden, dass sie automatisch während der Systeminitialisierung durchgeführt werden.

Es gibt zwei Typen von automatischen Mounts. Der erste Typ umfasst die Mounts, die erforderlich sind, um das System zu booten und auszuführen. Die Mounts dieser Dateisysteme werden explizit vom Bootprozess durchgeführt. Die Zeilengruppen für solche Dateisysteme in der Datei /etc/filesystems enthalten die Angabe `mount = automatic`. Der zweite Typ bezieht sich auf die automatischen Mounts, die vom Benutzer gesteuert werden. Die Mounts dieser Dateisysteme werden vom Script /etc/rc durchgeführt, wenn dieses den Befehl **mount all** absetzt. Die Zeilengruppen für die vom Benutzer gesteuerten automatischen Mounts enthalten die Angabe `mount = true` in der Datei /etc/filesystems.

Die Datei /etc/filesystems steuert die automatischen Mounts. Sie werden hierarchisch, Mountpunkt für Mountpunkt durchgeführt. Es ist auch möglich, die Mounts in einer bestimmten Reihenfolge anzuordnen, die geändert werden kann. Weitere Informationen zur Datei /etc/filesystems finden Sie im Abschnitt "/etc/filesystems".

Die Datei /etc/filesystems ist in Zeilengruppen, eine für jeden Mount, aufgebaut. Eine Zeilengruppe beschreibt die Attribute des entsprechenden Dateisystems und die Durchführung des Mount. Das System führt die Mounts der Dateisysteme in der Reihenfolge durch, in der die Dateisysteme in der Datei /etc/filesystems aufgeführt sind. Im Folgenden sehen Sie ein Beispiel für Zeilengruppen in der Datei /etc/filesystems:

```
/:
dev=/dev/hd4
vol="root"
mount=automatic
check=false
free=true
vfs=jfs
log=/dev/hd8
type=bootfs

/home:
dev=/dev/hd1
vfs=jfs
log=/dev/hd8
mount=true
check=true
vol="/home"
free=false

/usr:
/dev=/dev/hd2
vfs=jfs
log=/dev/hd8
mount=automatic
```

```
check=false
type=bootfs
vol="/usr"
free=false
```

Sie können die Datei `/etc/filesystems` editieren, um die Reihenfolge zu steuern, in der die Mounts durchgeführt werden. Wenn ein Mount fehlschlägt, können die folgenden Mounts, die in der Datei `/etc/filesystems` definiert sind, trotzdem durchgeführt werden. Sollte beispielsweise der Mount des Dateisystems `/home` fehlschlagen, wird der Mount des Dateisystems `/usr` trotzdem durchgeführt. Ursachen für das Fehlschlagen von Mounts können Schreibfehler, Abhängigkeits- oder Systemfehler sein.

Mountsicherheit für Workstations ohne Plattenspeicher:

Workstations ohne Plattenspeicher müssen in der Lage sein, Einheitengeräte dateien auf fernen Systemen abzulegen und auf diese zuzugreifen, damit ihre Verzeichnisse `/dev` über einen Server angehängt werden können. Da Server Einheitengeräte dateien, die für einen Client bestimmt sind, nicht von den für einen Server bestimmten Einheitengeräte dateien unterscheiden können, kann ein Benutzer auf dem Server unter Umständen mit den Geräte dateien der Client einheit auf die physischen Einheiten des Servers zugreifen.

Das Eigentumsrecht für ein nicht grafikfähiges Terminal (**TTY**) wird automatisch auf den Benutzer eingestellt, der das **TTY** verwendet. Wenn die Benutzer-IDs auf Client und Server nicht dieselben sind, kann ein nicht berechtigter Benutzer auf einem Server auf ein **TTY** zugreifen, das von einem anderen Benutzer auf dem Server verwendet wird.

Ein Benutzer, der auf einem Client berechtigt ist, kann Einheitengeräte dateien für die physischen Einheiten auf dem Server erstellen und so konfigurieren, dass sie keine Zugriffsberechtigung erfordern. Anschließend kann der Benutzer einen nicht berechtigten Account auf dem Server verwenden, um mit den neuen Einheitengeräte dateien auf die normalerweise geschützten Einheiten zuzugreifen.

Ein ähnliches Sicherheitsproblem besteht im Zusammenhang mit der Verwendung der Programme **setuid** und **setgid** auf Client und Server. Client ohne Plattenspeicher müssen in der Lage sein, die Programme **setuid** und **setgid** im normalen Betrieb auf dem Server auszuführen. Auch hier kann der Server nicht zwischen den für den Server und den für den Client bestimmten Programmen unterscheiden.

Außerdem stimmen möglicherweise die Benutzer-IDs und Gruppen-IDs zwischen Server und Client nicht überein, so dass Benutzer auf dem Server möglicherweise Programme mit Fähigkeiten ausführen können, die nicht für sie bestimmt sind.

Das Problem besteht darin, dass die Programme **setuid** und **setgid** sowie die Einheitengeräte dateien eigentlich nur auf der Maschine verwendbar sein sollten, auf der sie erstellt wurden.

Die Lösung des Problems besteht darin, mit dem Befehl **mount** Sicherheitsoptionen anzugeben, die die Möglichkeiten der Verwendung dieser Objekte beschränken. Diese Optionen können auch in den Zeilengruppen in der Datei `/etc/filesystems` verwendet werden.

Die Option **nosuid** des Befehls **mount** verhindert die Ausführung der Programme **setuid** und **setgid**, über die auf das angehängte Dateisystem zugegriffen wird. Diese Option wird für jedes Dateisystem verwendet, das auf einem bestimmten Host nur zur Verwendung durch einen anderen Host angehängt wird (z. B. exportiert für Clients ohne Plattenspeicher).

Die Option **nodev** des Befehls **mount** verhindert, dass Einheiten mit Einheitengeräte dateien geöffnet werden, über die auf das angehängte Dateisystem zugegriffen wird. Diese Option wird auch für alle Dateisysteme verwendet, die nur zur Verwendung durch einen anderen Host angehängt werden (z. B. exportiert für Clients ohne Plattenspeicher).

Im Allgemeinen haben Benutzer auf einem Server keinen Zugriff auf das Verzeichnis `/export`.

Verzeichnis /export/root exportieren

Das Verzeichnis /export/root muss mit Lese-/Schreibberechtigungen exportiert werden, und der Rootbenutzer des Servers muss Zugriff auf dieses Verzeichnis haben. Sie können dieses Verzeichnis jedoch auch mit den folgenden Optionen des Befehls **mount** anhängen:

| Eintrag | Beschreibung |
|---------------|---|
| nosuid | Verhindert, dass ein Benutzer auf dem Server die setuid -Programme des Clients ausführt. |
| nodev | Verhindert, dass ein Benutzer mit einer Einheitengeräte-datei des Clients auf die Servereinheiten zugreift. |

Eine Alternative zum Anhängen des Verzeichnisses /export/root mit diesen Optionen besteht darin, den Benutzern auf dem Server jeglichen Zugriff auf das Verzeichnis /export/root zu verweigern.

Verzeichnis /export/exec exportieren

Das Verzeichnis /export/exec wird nur mit Leseberechtigungen exportiert, und der Rootbenutzer muss Zugriff auf das Verzeichnis haben. Sie können dieses Verzeichnis jedoch auch mit den folgenden Optionen des Befehls **mount** anhängen:

| Eintrag | Beschreibung |
|---------------|---|
| nosuid | Verhindert, dass ein Benutzer auf dem Server die setuid -Programme des Clients ausführt. Wenn Sie das Serververzeichnis /usr exportieren, können Sie die Option nosuid nicht verwenden. |
| nodev | Verhindert, dass ein Benutzer mit einer Einheitengeräte-datei des Clients auf die Servereinheiten zugreift. |

Verzeichnis /export/share exportieren

Das Verzeichnis /export/share wird nur mit Leseberechtigungen exportiert, und der Rootbenutzer muss Zugriff auf das Verzeichnis haben. Da dieses Verzeichnis im Allgemeinen nur Daten (und keine ausführbaren Dateien oder Einheiten) enthält, müssen Sie die Sicherheitsoptionen für das Anhängen nicht verwenden.

Verzeichnis /export/home exportieren

Es gibt mehrere Möglichkeiten, das Verzeichnis /home eines Benutzers anzuhängen:

- Sie können das Verzeichnis /export/home/*Clienthostname* über das Clientverzeichnis /home anhängen. In diesem Fall hat der Client Lese-/Schreibberechtigungen und der Rootbenutzer Zugriff auf das Verzeichnis. Zur Gewährleistung der Systemsicherheit hängen Sie das Verzeichnis /export/home mit den folgenden Optionen des Befehls **mount** an:

| Eintrag | Beschreibung |
|---------------|---|
| nosuid | Verhindert, dass ein Benutzer auf dem Server die setuid -Programme des Clients ausführt. |
| nodev | Verhindert, dass ein Benutzer mit einer Einheitengeräte-datei des Clients auf die Servereinheiten zugreift. |

- Sie können das Verzeichnis /home auf dem Server über das Verzeichnis /home des Clients anhängen. In diesem Fall wird das Verzeichnis /home mit Lese-/Schreibberechtigungen ohne Rootzugriff exportiert. Zur Gewährleistung der Systemsicherheit hängen Sie das Verzeichnis /home auf dem Server und auf dem Client mit den Optionen **nosuid** und **nodev** des Befehls **mount** an.
- Alternativ können Sie auf dem Client jedes Verzeichnis /home/*Benutzername* auf dem Server über das Verzeichnis /home/*Benutzername* auf dem Client anhängen, so dass Benutzer sich an verschiedenen Maschinen anmelden können und trotzdem Zugriff auf ihre Ausgangsverzeichnisse haben. In diesem Fall werden die Verzeichnisse /home/*Benutzername* auf dem Server und auf den Clients mit den Optionen **nosuid** und **nodev** des Befehls **mount** angehängt.

Verzeichnis /export/swap exportieren

Exportieren Sie die Datei /export/swap/*Clienthostname* mit Lese-/Schreibberechtigungen und Rootzugriff. Es sind keine Sicherheitsmaßnahmen erforderlich. Benutzer auf dem Server haben keinen Zugriff auf die Dateien /export/swap/*Clienthostname*.

Mounts ohne Plattenspeicher:

Obwohl das Dateisystem einer Workstation ohne Plattenspeicher über das Verzeichnis `/exports` eines Servers angehängt wird (Mount), sieht das Dateisystem für die Workstation ohne Plattenspeicher wie das Dateisystem auf einer eigenständigen Maschine aus.

Im Folgenden wird die Beziehung zwischen einem Serverexport und einem Mountpunkt der Workstation ohne Plattenspeicher veranschaulicht.

| Serverexport | Import ohne Plattenspeicher |
|-------------------------------------|--|
| <code>/export/root/Hostname</code> | Stammdateisystem (<code>/</code>) |
| <code>/export/exec/SPOT-Name</code> | <code>/usr</code> |
| <code>/export/home/Hostname</code> | <code>/home</code> |
| <code>/export/share</code> | <code>/usr/share</code> |
| <code>/export/dump</code> | Wird vom Client ohne Plattenspeicher als Speicherauszugsbereich verwendet. |
| <code>/export/swap</code> | Wird von Clients ohne Plattenspeicher als ferner Paging-Bereich verwendet. |

Weitere Informationen zum Verzeichnis `/export` finden Sie im Abschnitt „Verzeichnis `/export`“ auf Seite 461.

Im Allgemeinen haben Benutzer auf einem Server keinen Zugriff auf das Verzeichnis `/export`.

Verzeichnis `/export/root` exportieren

Das Verzeichnis `/export/root` muss mit Lese-/Schreibberechtigungen exportiert werden, und der Rootbenutzer des Servers muss Zugriff auf dieses Verzeichnis haben. Sie können dieses Verzeichnis jedoch auch mit den folgenden Optionen des Befehls **mount** anhängen:

| Eintrag | Beschreibung |
|---------------|---|
| nosuid | Verhindert, dass ein Benutzer auf dem Server die setuid -Programme des Clients ausführt. |
| nodev | Verhindert, dass ein Benutzer mit einer Einheitengeräte-datei des Clients auf die Servereinheiten zugreift. |

Eine Alternative zum Anhängen des Verzeichnisses `/export/root` mit diesen Optionen besteht darin, den Benutzern auf dem Server jeglichen Zugriff auf das Verzeichnis `/export/root` zu verweigern.

Verzeichnis `/export/exec` exportieren

Das Verzeichnis `/export/exec` wird nur mit Leseberechtigungen exportiert, und der Rootbenutzer muss Zugriff auf das Verzeichnis haben. Sie können dieses Verzeichnis jedoch auch mit den folgenden Optionen des Befehls **mount** anhängen:

| Eintrag | Beschreibung |
|---------------|--|
| nosuid | Verhindert, dass ein Benutzer auf dem Server die setuid -Programme des Clients ausführt. Wenn Sie das Serververzeichnis <code>/usr</code> exportieren, können Sie die Option nosuid nicht verwenden. |
| nodev | Verhindert, dass ein Benutzer mit einer Einheitengeräte-datei des Clients auf die Servereinheiten zugreift. |

Verzeichnis `/export/share` exportieren

Das Verzeichnis `/export/share` wird nur mit Leseberechtigungen exportiert, und der Rootbenutzer muss Zugriff auf das Verzeichnis haben. Da dieses Verzeichnis im Allgemeinen nur Daten (und keine ausführbaren Dateien oder Einheiten) enthält, müssen Sie die Sicherheitsoptionen für das Anhängen nicht verwenden.

Verzeichnis `/export/home` exportieren

Es gibt mehrere Möglichkeiten, das Verzeichnis `/home` eines Benutzers anzuhängen:

- Sie können das Verzeichnis `/export/home/Clienthostname` über das Clientverzeichnis `/home` anhängen. In diesem Fall hat der Client Lese-/Schreibberechtigungen und der Rootbenutzer Zugriff auf das Verzeichnis. Zur Gewährleistung der Systemsicherheit hängen Sie das Verzeichnis

/export/home mit den folgenden Optionen des Befehls **mount** an:

| Eintrag | Beschreibung |
|---------------|---|
| nosuid | Verhindert, dass ein Benutzer auf dem Server die setuid -Programme des Clients ausführt. |
| nodev | Verhindert, dass ein Benutzer mit einer Einheitengeräte-datei des Clients auf die Servereinheiten zugreift. |

- Sie können das Verzeichnis /home auf dem Server über das Verzeichnis /home des Clients anhängen. In diesem Fall wird das Verzeichnis /home mit Lese-/Schreibberechtigungen ohne Rootzugriff exportiert. Zur Gewährleistung der Systemsicherheit hängen Sie das Verzeichnis /home auf dem Server und auf dem Client mit den Optionen **nosuid** und **nodev** des Befehls **mount** an.
- Alternativ können Sie auf dem Client jedes Verzeichnis /home/*Benutzername* auf dem Server über das Verzeichnis /home/*Benutzername* auf dem Client anhängen, so dass Benutzer sich an verschiedenen Maschinen anmelden können und trotzdem Zugriff auf ihre Ausgangsverzeichnisse haben. In diesem Fall werden die Verzeichnisse /home/*Benutzername* auf dem Server und auf den Clients mit den Optionen **nosuid** und **nodev** des Befehls **mount** angehängt.

Verzeichnis /export/dump exportieren

Exportieren Sie das Verzeichnis /export/dump/*Clienthostname* mit Lese-/Schreibberechtigungen und Rootzugriff. Benutzer auf dem Server haben keinen Zugriff auf die Dateien /export/dump/*Clienthostname*.

Verzeichnis /export/swap exportieren

Exportieren Sie die Datei /export/swap/*Clienthostname* mit Lese-/Schreibberechtigungen und Rootzugriff. Es sind keine Sicherheitsmaßnahmen erforderlich. Benutzer auf dem Server haben keinen Zugriff auf die Dateien /export/swap/*Clienthostname*.

Dateisystemtypen

AIX unterstützt mehrere Dateisystemtypen.

Im Folgenden sind einige dieser Dateisystemtypen beschrieben.

Journalized File System (JFS) und Enhanced Journalized File System (JFS2)

Unterstützt die vollständige Dateisystemsemantik. Diese Dateisysteme verwenden Verfahren zur Journalführung für Datenbanken, um die strukturelle Konsistenz zu bewahren. Dadurch werden Beschädigungen am Dateisystem bei abnormalen Systemstopps verhindert.

Jedes JFS oder JFS2 befindet sich auf einem eigenen logischen Datenträger. Das Betriebssystem hängt das Dateisystem während der Initialisierung an. Diese Konfiguration mit mehreren Dateisystemen ist hilfreich für Systemverwaltungsfunktionen wie das Sichern, Wiederherstellen und Reparieren, weil sie einen Teil des Dateibaums isoliert, damit sie ihn bearbeiten können.

JFS ist der Basisdateisystemtyp, der die vollständige Gruppe der Dateisystembefehle unterstützt.

JFS2 ist der Basisdateisystemtyp, der die vollständige Gruppe der Dateisystembefehle unterstützt.

Ein Unterschied zwischen JFS und JFS2 ist der, dass JFS2 auch große Dateien und große Dateisysteme unterstützt.

Network File System (NFS)

Ein verteiltes Dateisystem, das es Benutzern ermöglicht, auf Dateien und Verzeichnisse auf fernen Computern zuzugreifen und diese Dateien und Verzeichnisse so zu verwenden, als wären es lokale Dateien und Verzeichnisse. Benutzer können beispielsweise die Betriebssystembefehle verwenden, um ferne Dateien und Verzeichnisse zu erstellen, zu entfernen, zu lesen, zu schreiben und Dateiattribute zu ändern.

CD-ROM File System (CDRFS)

Ermöglicht den Zugriff auf den Inhalt einer CD-ROM über die herkömmlichen Dateischnittstellen (open, read und close).

DVD-ROM File System (UDFS)

Ermöglicht den Zugriff auf den Inhalt einer DVD über die herkömmlichen Dateischnittstellen.

Zugehörige Informationen:

Network file system

JFS und JFS2:

Das Journaled File System (JFS) und das Enhanced Journaled File System (JFS2) sind in das Basisbetriebssystem integriert. Beide Dateisystemtypen verknüpfen ihre Datei- und Verzeichnisdaten mit der Struktur, die vom Logical Volume Manager in AIX für das Speichern und den Abruf von Daten verwendet wird.

Ein Unterschied besteht darin, dass JFS2 für die Unterstützung eines 64-Bit-Kernels und größerer Dateien ausgelegt ist.

In den folgenden Abschnitten werden diese Dateisysteme beschrieben. Sofern nicht anders angegeben, beziehen sich die folgenden Abschnitte gleichermaßen auf JFS und JFS2.

JFS- und JFS2-Funktionen:

Enhanced Journaled File System (JFS2) ist ein Dateisystem, das die Möglichkeit bietet, sehr viel größere Dateien zu speichern, als es im vorhandenen Journaled File System (JFS) möglich ist.

Sie können entweder JFS oder JFS2 implementieren. JFS2 ist das Standarddateisystem in AIX 6.1.

Anmerkung: Anders als das Dateisystem JFS lässt das Dateisystem JFS2 die Verwendung der API `link()` für Dateien des Typs Verzeichnis nicht zu. Diese Einschränkung kann dazu führen, dass einige Anwendungen, die in einem JFS-Dateisystem ordnungsgemäß funktionieren, in einem JFS2-Dateisystem fehlschlagen.

Die folgende Tabelle enthält eine Zusammenfassung der JFS- und JFS2-Funktionen:

| Funktionen | JFS2 | JFS |
|---------------------------|---|---|
| Fragment- und Blockgrößen | Blockgrößen (Bytes): 512, 1024, 2048, 4096
Maximale Dateisystemgröße in Terabytes (TB): 4, 8, 16, 32 | Fragmentgrößen (Bytes): 512, 1024, 2048, 4096
Maximale Dateisystemgröße in Gigabytes (GB): 128, 256, 512, 1024 |
| Maximale Dateisystemgröße | 32 TB | 1 TB |
| Mindestdateisystemgröße | 16 MB | Nicht zutreffend |
| Maximale Dateigröße | 16 TB | Ungefähr 63,876 GB |
| Anzahl der I-Nodes | Dynamisch, begrenzt durch Plattenspeicherplatz | Fest, Definition bei Erstellung des Dateisystems |
| Verzeichnisorganisation | B-Tree | Linear |
| Komprimierung | Nein | Ja |
| Kontingente | Ja | Ja |
| Fehlerprotokollierung | Ja | Ja |

Anmerkung:

1. Die maximale Dateigröße und die maximale Dateisystemgröße sind auf (1 TB - (Größe der physischen Partition)) begrenzt, wenn ein 32-Bit-Kernel verwendet wird. Wenn die Größe der physischen Partition für die Datenträgergruppe beispielsweise 64 MB beträgt, ist die maximale Dateisystemgröße (1 TB - 64

MB) = (1048576 MB - 64 MB) = 1048512 MB. Dies ist auf eine Beschränkung der maximalen Größe eines logischen Datenträgers bei Verwendung des 32-Bit-Kernels zurückzuführen.

2. JFS2 unterstützt das Standardprotokollierungsschema von AIX. Weitere Informationen zur Fehlerprotokollierung in AIX finden Sie im Abschnitt "Error-Logging Overview" in der Veröffentlichung *General Programming Concepts: Writing and Debugging Programs*.

Segmentierung des Plattenspeicherplatzes in JFS und JFS2:

Viele UNIX-Dateisysteme ordnen den logischen Blöcken, die für die logische Unterteilung von Dateien und Verzeichnissen verwendet werden, nur zusammenhängenden Speicherplatz in Einheiten identischer Größe zu. Diese Zuordnungseinheiten werden gewöhnlich als *Plattenblöcke* bezeichnet. Ein Plattenblock wird ausschließlich dazu verwendet, um die Daten, die in einem logischen Block einer Datei oder eines Verzeichnisses enthalten sind, zu speichern.

Durch die Verwendung relativ großer logischer Blöcke (z. B. 4096 Bytes) und die Zuordnung von Plattenblöcken identischer Größe zu logischen Blöcken kann die Anzahl der Platten-E/A-Operationen, die in einer einzigen Dateisystemoperation ausgeführt werden müssen, reduziert werden. Die Daten einer Datei oder eines Verzeichnisses werden in wenigen großen Plattenblöcken anstatt in vielen kleinen Plattenblöcken auf der Platte gespeichert. Einer Datei mit einer Größe von 4096 Bytes oder weniger wird beispielsweise ein einziger 4096-Byte-Plattenblock zugeordnet, wenn die Größe der logischen Blöcke 4096 Bytes beträgt. Eine Lese- oder Schreiboperation muss deshalb nur eine einzige Platten-E/A-Operation ausführen, um auf die Daten auf der Platte zuzugreifen. Wenn die Größe der logischen Blöcke kleiner ist und mehr als eine Zuordnungseinheit für dieselbe Datenmenge benötigt, müssen für den Zugriff auf die Daten mehrere Platten-E/A-Operationen ausgeführt werden. Durch einen großen logischen Block und eine identische Plattenblockgröße können die Aktivitäten für die Zuordnung von Plattenspeicherplatz, die ausgeführt werden müssen, wenn neue Daten zu Dateien und Verzeichnissen hinzugefügt werden, reduziert werden, weil größere Plattenblöcke mehr Daten speichern können.

Die Einschränkung der Zuordnungseinheit für den Plattenspeicherplatz auf die Größe der logischen Blöcke kann jedoch dazu führen, dass Plattenspeicherplatz in einem Dateisystem verschwendet wird, das zahlreiche kleine Dateien und Verzeichnisse enthält. Wenn nur ein Teil eines logischen Blocks durch eine Datei oder ein Verzeichnis belegt wird, wird Plattenspeicherplatz verschwendet. Da ein nur teilweise genutzter logischer Block immer weniger Plattenspeicherplatz nutzt, als ein logischer Block reserviert, ist mehr Plattenspeicherplatz als benötigt reserviert. Der verbleibende Teil bleibt ungenutzt, weil keine andere Datei und kein anderes Verzeichnis ihren bzw. seinen Inhalt in bereits zugeordneten Plattenspeicherplatz schreiben kann. Der verschwendete Plattenspeicherplatz kann für Dateisysteme, die sehr viele kleine Dateien und Verzeichnisse enthält, insgesamt sehr hoch werden.

Das Journaled File System (JFS) teilt den Plattenspeicherplatz in Zuordnungseinheiten ein, so genannte *Fragmente*. Das Enhanced Journaled File System (JFS2) segmentiert den Plattenspeicherplatz in *Blöcke*. Die Zielsetzung ist jeweils dieselbe: eine effiziente Speicherung von Daten.

JFS-Fragmente sind kleiner als die Standardplattenzuordnungsgröße von 4096 Bytes. Die Verwendung von Fragmenten verringert die Verschwendung von Plattenspeicherplatz, weil die Daten effizienter in den logischen Blöcken einer Datei oder eines Verzeichnisses gespeichert werden können. Das funktionale Verhalten der JFS-Fragmentunterstützung basiert auf der von Berkeley Software Distribution (BSD) bereitgestellten Fragmentunterstützung.

JFS2 unterstützt mehrere Blockgrößen für das Dateisystem: 512, 1024, 2048 und 4096. Die Verwendung einer kleineren Blockgröße verringert die Verschwendung von Plattenspeicherplatz, weil die Daten effizienter in den logischen Blöcken einer Datei oder eines Verzeichnisses gespeichert werden können. Kleinere Blockgrößen führen außerdem zu einem zusätzlichen Betriebsaufwand. Die Blockgröße für ein JFS2 wird während der Erstellung des Dateisystems angegeben. Unterschiedliche Dateisysteme können unterschiedliche Blockgrößen haben, aber in einem Dateisystem kann jeweils nur eine Blockgröße verwendet werden.

Zugehörige Konzepte:

„Datenkomprimierung in JFS“ auf Seite 497

JFS unterstützt fragmentierte und komprimierte Dateisysteme, die Plattenspeicherplatz einsparen, indem zugelassen wird, dass ein logischer Block in Einheiten oder "Fragmenten", die kleiner sind als die vollständige Blockgröße von 4096 Bytes, auf der Platte gespeichert werden.

JFS-Fragmente:

In JFS ist die Zuordnungseinheit für Plattenspeicherplatz ein *Fragment*. Ein Fragment kann kleiner sein als 4096 Bytes, die Größe für logische Blöcke.

Wenn Fragmente verwendet werden, die kleiner sind als 4096 Bytes, können die in einem logischen Teilblock enthaltenen Daten effizienter gespeichert werden, weil nur so viele Fragmente verwendet werden, wie für die Speicherung der Daten erforderlich sind. Beispielsweise könnte einem logischen Teilblock mit nur 500 Bytes ein Fragment mit 512 Bytes (bei einer angenommenen Fragmentgröße von 512 Bytes) zugeordnet werden, was die Menge verschwendeten Speicherplatzes erheblich reduzieren würde. Wenn die Speicheranforderungen eines logischen Teilblocks zunehmen, können weitere Fragmente zugeordnet werden.

Die Fragmentgröße für ein Dateisystem wird während der Erstellung des Dateisystems angegeben. Die zulässigen Fragmentgrößen für Journaled File Systems (JFS) sind 512, 1024, 2048 und 4096 Bytes. Unterschiedliche Dateisysteme können unterschiedliche Fragmentgrößen haben, aber in einem Dateisystem kann jeweils nur eine Fragmentgröße verwendet werden. Es können auch mehrere Fragmentgrößen auf einem einzelnen System (Maschine) verwendet werden, so dass die Benutzer für jedes Dateisystem die jeweils geeignete Fragmentgröße auswählen können.

Die JFS-Fragmentunterstützung liefert eine Sicht, in der das Dateisystem als Reihe zusammenhängender Fragmente und nicht als Reihe zusammenhängender Plattenblöcke dargestellt wird. Zur Gewährleistung effizienter Plattenoperationen wird der Plattenspeicherplatz jedoch häufig in Einheiten von 4096 Bytes zugeordnet, so dass die Plattenblöcke bzw. Zuordnungseinheiten dieselbe Größe haben wie die logischen Blöcke. Die Zuordnung von Plattenblöcken kann in diesem Fall als Zuordnung von 4096 Bytes zusammenhängender Fragmente betrachtet werden.

Betriebsaufwand (zusätzliche Suchvorgänge auf der Platte, Datenübertragungen und Zuordnungsaktivität) und bessere Nutzung des Plattenspeicherplatzes erhöhen sich, je niedriger die Fragmentgröße für ein Dateisystem ist. Um eine optimale Balance zwischen erhöhtem Aufwand und erhöhter Nutzbarkeit von Plattenspeicherplatz zu erzielen, müssen die folgenden Faktoren bei der JFS-Fragmentunterstützung berücksichtigt werden:

- Sofern möglich, werden bei der Zuordnung von Plattenspeicherplatz 4096 Bytes an Fragmenten für eine Datei oder die logischen Blöcke eines Verzeichnisses verwendet.
- Nur logischen Teilblöcken für Dateien oder Verzeichnisse mit weniger als 32 KB können weniger als 4096 Bytes an Fragmenten zugeordnet werden.

Wenn die Dateien und Verzeichnisse in einem Dateisystem die Größe von 32 KB überschreiten, schwindet der Vorteil, weniger als 4096 Bytes Plattenspeicherplatz für logische Teilblöcke zuzuordnen. Die Einsparungen an Plattenspeicherplatz, ausgedrückt als prozentualer Anteil des Gesamtspeicherplatzes im Dateisystem, werden gering, während die zusätzlichen Leistungseinbußen für die Verwaltung kleiner Zuordnungseinheiten für den Plattenspeicherplatz konstant bleiben. Da kleinere Zuordnungseinheiten als 4096 Bytes bei kleinen Dateien und Verzeichnissen für die effektivste Nutzung des Plattenspeicherplatzes sorgen, werden den logischen Blöcken von Dateien und Verzeichnissen, die größer-gleich 32 KB sind, immer 4096 Bytes an Fragmenten zugeordnet. Jedem logischen Teilblock, der zu einer solchen großen Datei oder einem solch großen Verzeichnis gehört, werden ebenfalls 4096 Bytes an Fragmenten zugeordnet.

JFS2-Blöcke:

Enhanced Journaled File System segmentiert den Plattenspeicherplatz in *Blöcke*. JFS2 unterstützt mehrere Blockgrößen für das Dateisystem: 512, 1024, 2048 und 4096.

Unterschiedliche Dateisysteme können unterschiedliche Blockgrößen haben, aber in einem Dateisystem kann jeweils nur eine Blockgröße verwendet werden.

Die Verwendung einer kleineren Blockgröße verringert die Verschwendung von Plattenspeicherplatz, weil die Daten effizienter in den logischen Blöcken einer Datei oder eines Verzeichnisses gespeichert werden können. Kleinere Blockgrößen können außerdem zu einem zusätzlichen Betriebsaufwand führen. Außerdem müssen Einheitsreiber eine Adressierbarkeit von Plattenblöcken unterstützen, die kleiner-gleich der Blockgröße des Dateisystems ist.

Da der Plattenspeicherplatz für ein Dateisystem mit einer anderen Blockgröße als 4096 Bytes in kleineren Einheiten zugeordnet wird, kann die Zuordnungsaktivität zunehmen, wenn die Größe von Dateien oder Verzeichnissen wiederholt heraufgesetzt wird. Beispielsweise führt eine Schreiboperation, die die Größe einer Datei mit der Länge null um 512 Bytes erhöht, dazu, dass der Datei bei vorausgesetzter Blockgröße von 512 Bytes ein Block zugeordnet wird. Wenn die Dateigröße bei einer weiteren Schreiboperation um weitere 512 Bytes erhöht wird, muss der Datei ein zusätzlicher Block zugeordnet werden. Wenn Sie dieses Beispiel auf ein Dateisystem mit 4096-Byte-Blöcken anwenden, findet eine Zuordnung von Plattenspeicherplatz nur einmal und zwar im Rahmen der ersten Schreiboperation statt. Bei der zweiten Schreiboperation wird kein weiterer Plattenspeicherplatz zugeordnet, weil die erste Zuordnung eines 4096-Byte-Blocks ausreicht, um die Daten aufzunehmen, die durch die zweite Schreiboperation hinzugefügt werden.

Die Blockgröße des Dateisystems wird während der Erstellung des Dateisystems mit System Management Interface Tool (SMIT) oder den Befehlen **crfs** und **mkfs** angegeben. Welche Blockgröße für das Dateisystem ausgewählt werden sollte, richtet sich nach der geplanten Größe der Dateien, die im Dateisystem enthalten sind.

Der Wert für die Blockgröße des Dateisystems kann mit System Management Interface Tool (SMIT) oder dem Befehl **lsfs** angegeben werden. Für Anwendungsprogramme kann die Blockgröße des Dateisystems mit der Subroutine **statfs** angegeben werden.

Blöcke dienen als Basiseinheit für die Zuordnung von Plattenspeicherplatz, und der Zuordnungsstatus jedes einzelnen Blocks in einem Dateisystem wird in der Blockzuordnungsübersicht für das Dateisystem aufgezeichnet. Für Dateisysteme mit einer kleineren Blockgröße als 4096 Bytes ist unter Umständen mehr virtueller Speicher und mehr Plattenspeicherplatz im Dateisystem erforderlich, um die Blockzuordnungsübersichten zu speichern.

Variable Anzahl von I-Nodes:

Wenn Sie den Plattenspeicherplatz in Einheiten mit weniger als 4096 Bytes segmentieren, wird dadurch zwar die Belegung des Plattenspeicherplatzes optimiert, aber die Anzahl kleiner Dateien und Verzeichnisse, die in einem Dateisystem gespeichert werden können, nimmt zu.

Der Plattenspeicherplatz ist jedoch nur eine der Dateisystemressourcen, die Dateien und Verzeichnisse benötigen. Jede Datei und jedes Verzeichnis benötigt außerdem einen Platten-I-Node.

JFS und I-Nodes:

Bei JFS können Sie die Anzahl der Platten-I-Nodes, die in einem Dateisystem erstellt werden, angeben, falls mehr oder weniger als die Standardanzahl benötigt wird.

Die Anzahl der Platten-I-Nodes wird bei der Erstellung des Dateisystems in *Number of Bytes per I-Node* (Anzahl der Bytes pro I-Node) oder *NBPI* angegeben. Wenn Sie beispielsweise einen NBPI-Wert von 1024 angeben, wird pro 1024 Bytes Plattenspeicherplatz im Dateisystem ein Platten-I-Node erstellt. Anders ausgedrückt, ein kleiner NBPI-Wert (z. B. 512) führt zu einer größeren Anzahl von I-Nodes, während ein hoher NBPI-Wert (z. B. 16384) eine kleinere Anzahl von I-Nodes zur Folge hat.

Pro NBPI Bytes in der Zuordnungsgruppe, die dem Dateisystem zur Verfügung steht, wird in JFS ein I-Node angelegt. Die Gesamtanzahl der I-Nodes in einem Dateisystem beschränkt die Gesamtanzahl der Dateien und die Gesamtgröße des Dateisystems. Eine Zuordnungsgruppe kann teilreserviert sein, obwohl die gesamte Anzahl der I-Nodes für die Zuordnungsgruppe reserviert ist. NBPI ist umgekehrt proportional zur Gesamtanzahl der I-Nodes in einem Dateisystem.

JFS beschränkt alle Dateisysteme auf 16 Millionen (2^{24}) I-Nodes.

Die Gruppe zulässiger NBPI-Werte variiert mit der Größe der Zuordnungsgruppe (*agsize*). Der Standardwert sind 8 MB. Die zulässigen NBPI-Werte sind 512, 1024, 2048, 4096, 8192 und 16384 bei einer Zuordnungsgruppengröße (*agsize*) von 8 MB. Es kann ein höherer *agsize*-Wert verwendet werden. Die zulässigen Werte für *agsize* sind 8, 16, 32 und 64. Der Bereich zulässiger NBPI-Werte nimmt maßstäblich mit steigendem *agsize*-Wert zu. Wenn der *agsize*-Wert auf 16 MB verdoppelt wird, verdoppelt sich auch der Bereich der NBPI-Werte: 1024, 2048, 4096, 8193, 16384 und 32768.

Die Fragmentgröße und der NBPI-Wert werden während der Erstellung des Dateisystems mit System Management Interface Tool (SMIT) oder den Befehlen **crfs** und **mkfs** angegeben. Die Entscheidung über die Fragmentgröße und die Anzahl der I-Nodes für das Dateisystem richtet sich nach der geplanten Anzahl und Größe der Dateien im Dateisystem.

Sie können die Fragmentgröße und den NBPI-Wert mit System Management Interface Tool (SMIT) oder dem Befehl **lsfs** angeben. Für Anwendungsprogramme können Sie die Fragmentgröße für das Dateisystem mit der Subroutine **statfs** angeben.

JFS2 und I-Nodes:

JFS2 ordnet I-Nodes bei Bedarf zu.

Wenn in einem Dateisystem Platz für zusätzliche I-Nodes ist, werden diese automatisch reserviert. Deshalb ist die Anzahl der verfügbaren I-Nodes durch die Größe des Dateisystems selbst beschränkt.

Größenbegrenzungen in JFS und JFS2:

Sie definieren die maximale Größe für ein JFS, wenn Sie das Dateisystem erstellen. Die Definition der Größe für ein JFS richtet sich nach mehreren wichtigen Faktoren.

Die empfohlene maximale Größe für ein JFS2 sind 16 TB. Die Mindestgröße für ein JFS2 sind 16 MB.

Obwohl Dateisysteme, die kleinere Zuordnungseinheiten als 4096 Bytes verwenden, wesentlich weniger Plattenspeicherplatz erfordern als Dateisysteme, die die Standardzuordnungseinheit von 4096 Bytes verwenden, kann die Verwendung kleinerer Fragmente zu Lasten der Leistung gehen.

Der Zuordnungsstatus jedes Fragments (JFS) bzw. Blocks (JFS2) in einem Dateisystem wird in der Zuordnungsübersicht des Dateisystems aufgezeichnet. Für Dateisysteme mit einer kleineren Fragment- bzw. Blockgröße als 4096 Bytes ist unter Umständen mehr virtueller Speicher und mehr Plattenspeicherplatz im Dateisystem erforderlich, um die Zuordnungsübersichten zu speichern.

Da der Plattenspeicherplatz für ein Dateisystem mit einer anderen Fragment- (JFS) bzw. Blockgröße (JFS2) als 4096 Bytes in kleineren Einheiten zugeordnet wird, kann die Zuordnungsaktivität zunehmen, wenn die Größe von Dateien oder Verzeichnissen wiederholt heraufgesetzt wird. Beispielsweise führt eine Schreiboperation, die die Größe einer Datei mit der Länge null um 512 Bytes erhöht, dazu, dass der Datei je nach Dateisystemtyp ein 512-Byte-Fragment bzw. -Block zugeordnet wird. Wenn die Dateigröße bei einer weiteren Schreiboperation um weitere 512 Bytes erhöht wird, muss der Datei ein zusätzliches Fragment bzw. ein zusätzlicher Block zugeordnet werden. Wenn Sie dieses Beispiel auf ein Dateisystem mit 4096-Byte-Fragmenten bzw. -Blöcken anwenden, findet eine Zuordnung von Plattenspeicherplatz nur einmal und zwar im Rahmen der ersten Schreiboperation statt. Bei der zweiten Schreiboperation muss kein

weiterer Plattenspeicherplatz zugeordnet werden, weil die erste 4096-Byte-Zuordnung ausreicht, um die Daten aufzunehmen, die durch die zweite Schreiboperation hinzugefügt werden. Die Zuordnungsaktivität kann minimiert werden, wenn die Dateien um jeweils 4096 Bytes vergrößert werden.

Ein Aspekt, der sich auf die Größe bezieht, ist die Größe des Dateisystemprotokolls.

Bei JFS verwenden mehrere Dateisysteme meistens ein gemeinsames Protokoll, das mit einer Größe von 4 MB konfiguriert ist. Beispielsweise verwenden nach einer Erstinstallation alle Dateisysteme in der Stammdatenträgergruppe den logischen Datenträger hd8 als gemeinsames JFS-Protokoll. Die Standardpartitionsgröße für logische Datenträger sind 4 MB, und die Standardprotokollgröße ist eine Partition. Deshalb enthält die Stammdatenträgergruppe in der Regel ein JFS-Protokoll mit einer Größe von 4 MB. Wenn Dateisysteme größer sind als 2 GB oder wenn der Gesamtspeicherplatz eines Dateisystems, das ein einziges Protokoll verwendet, 2 GB überschreitet, ist die Standardprotokollgröße möglicherweise nicht ausreichend. In jedem Fall steigen die Protokollgrößen maßstäblich mit der Zunahme der Dateisystemgröße. Wenn die Größe des logischen Datenträgers mit dem Protokoll geändert wird, muss der Befehl **logform** ausgeführt werden, um das Protokoll zu reinitialisieren, bevor neuer Speicherplatz verwendet werden kann. Das JFS-Protokoll ist auf eine maximale Größe von 256 MB beschränkt.

Es gibt eine zweckmäßige Begrenzung für die Größe der kombinierten Dateisysteme, die ein einziges JFS-Protokoll unterstützen kann. Als Richtlinie wird eine Billion Bytes Gesamtkapazität der Dateisysteme als Grenzwert für ein einziges JFS-Protokoll empfohlen. Wenn dieser Richtwert überschritten wird, seine Überschreitung bevorsteht oder Fehler wegen mangelnden Speicherplatzes bei der Ausführung des Befehls **logredo** auftreten (der vom Befehl **fsck** aufgerufen wird), fügen Sie ein zusätzliches JFS-Protokoll hinzu, und verteilen Sie anschließend die Last auf die beiden JFS-Protokolldateien.

Bei JFS2 verwenden mehrere Dateisysteme meistens ein gemeinsames Protokoll. Wenn Dateisysteme größer sind als 2 GB oder wenn der Gesamtspeicherplatz eines Dateisystems, das ein einziges Protokoll verwendet, 2 GB überschreitet, ist die Standardprotokollgröße möglicherweise nicht ausreichend. In jedem Fall können Sie die Protokollgröße bei Zunahme der Dateisystemgröße heraufsetzen oder ein zusätzliches JFS2-Protokoll hinzufügen und anschließend die Last auf die beiden JFS2-Protokolldateien verteilen.

JFS-Größenbegrenzungen:

Die maximale JFS-Größe wird definiert, wenn das Dateisystem erstellt wird. Der NBPI-Wert, die Fragmentgröße und die Größe der Zuordnungsgruppe tragen maßgeblich zu dieser Entscheidung bei.

Die maximale Größe des Dateisystems entspricht der kleinsten der folgenden Größen:

$$NBPI * 2^{24}$$

und

$$Fragmentgröße * 2^{28}$$

Wenn Sie beispielsweise einen NBPI-Faktor von 512 auswählen, ist die Größe des Dateisystems auf 8 GB beschränkt ($512 * 2^{24} = 8 \text{ GB}$). JFS unterstützt die NBPI-Werte 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536 und 131072.

JFS beschränkt alle Dateisysteme auf 16 Millionen (2^{24}) I-Nodes.

Pro NBPI Bytes in der Zuordnungsgruppe, die dem Dateisystem zur Verfügung steht, wird ein I-Node angelegt. Eine Zuordnungsgruppe kann teilreserviert sein, obwohl die gesamte Anzahl der I-Nodes für die Zuordnungsgruppe reserviert ist. NBPI ist umgekehrt proportional zur Gesamtanzahl der I-Nodes in einem Dateisystem.

Das JFS teilt den Speicherplatz des Dateisystems in Gruppierungen von I-Nodes und Plattenblöcken für Benutzerdaten auf. Diese Gruppierungen werden als Zuordnungsgruppen bezeichnet. Die Größe der Zuordnungsgruppe kann beim Erstellen des Dateisystems angegeben werden. Die gültigen Größen für Zu-

ordnungsgruppen sind 8 MB, 16 MB, 32 MB und 64 MB. Jede Zuordnungsgruppengröße hat einen zugehörigen NBPI-Bereich. Die Bereiche sind in der folgenden Tabelle definiert:

Zuordnungsgruppe

| Größe in MB | Zulässige NBPI-Werte |
|-------------|---|
| 8 | 512, 1024, 2048, 4096, 8192, 16384 |
| 16 | 1024, 2048, 4096, 8192, 16384, 32768 |
| 32 | 2048, 4096, 8192, 16384, 32768, 65536 |
| 64 | 4096, 8192, 16384, 32768, 65536, 131072 |

JFS unterstützt vier Fragmentgrößen: 512-, 1024-, 2048- und 4096-Byte-Einheiten jeweils zusammenhängenden Plattenspeicherplatzes. JFS verwaltet Fragmentadressen in I-Nodes und Zeigerblöcken als 28-Bit-Zahlen. Jedes Fragment muss mit einer Zahl von 0 bis (2^{28}) adressierbar sein.

JFS2-Größenbegrenzungen:

Tests haben gezeigt, dass extrem große JFS2-Dateisysteme, die sehr große Dateien enthalten, einfacher zu verwalten sind als Dateisysteme, die eine große Anzahl kleiner Dateien enthalten. Wenn ein großes Dateisystem viele kleine Dateien enthält, dauert die Ausführung des Befehls **fsck** und anderer Verwaltungstasks für Dateisysteme sehr lange.

Die folgenden Größenbegrenzungen werden empfohlen:

| Eintrag | Beschreibung |
|--|--------------|
| Maximale Größe für ein JFS2-Dateisystem: | 32 TB |
| Maximale Dateigröße in JFS2: | 16 TB |
| Mindestgröße für ein JFS2-Dateisystem: | 16 MB |

Fragmentierung freien Speicherplatzes in einem JFS:

Wenn in JFS-Dateisystemen kleinere Fragmente als 4096 Bytes verwendet werden, kann dies zu einer größeren Fragmentierung des freien Speicherplatzes auf der Platte führen.

Stellen Sie sich beispielsweise einen Bereich der Platte vor, der in acht Fragmente mit jeweils 512 Bytes eingeteilt ist. Nehmen Sie an, dass unterschiedliche Dateien, die jeweils 512 Bytes erfordern, in das erste, vierte, fünfte und siebte Fragment in diesem Bereich der Platte geschrieben wurden und dass das zweite, dritte, sechste und achte Fragment leer sind. Obwohl vier Fragmente, die insgesamt 2048 Bytes des Plattenspeicherplatzes darstellen, frei sind, wird kein logischer Teilblock, der vier Fragmente (oder 2048 Bytes) erfordert, für diese freien Fragmente zugeordnet, weil die Fragmente in einer einzelnen Zuordnung benachbart sein müssen.

Da die für die logischen Blöcke einer Datei oder eines Verzeichnisses zugeordneten Fragmente benachbart sein müssen, kann die Fragmentierung des freien Speicherplatzes dazu führen, dass eine Dateisystemoperation, die neuen Plattenspeicherplatz anfordert, scheitert, selbst wenn der insgesamt verfügbare freie Speicherplatz groß genug ist, um die Operation zu unterstützen. Beispielsweise erfordert eine Schreiboperation, die eine Datei mit der Länge null um einen logischen Block erweitert, 4096 Bytes benachbarten Plattenspeicherplatzes für die Zuordnung. Wenn der freie Speicherplatz im Dateisystem fragmentiert ist und sich aus 32 nicht benachbarten 512-Byte-Fragmenten oder insgesamt 16 KB freiem Plattenspeicherplatz zusammensetzt, scheitert die Schreiboperation, weil keine acht benachbarten Fragmente (oder 4096 Bytes benachbarten Plattenspeicherplatzes) für die Schreiboperation verfügbar sind.

Ein JFS-Dateisystem mit einer nicht verwaltbaren Menge fragmentierten freien Speicherplatzes kann mit dem Befehl **defragfs** defragmentiert werden. Die Ausführung des Befehls **defragfs** hat eine positive Wirkung auf die Leistung.

Dateien mit freien Bereichen:

Eine Datei ist eine Folge indexierter Blöcke. Blöcke werden über den I-Node dem logischen Offset der Datei zugeordnet, die sie darstellen.

Eine Datei, die mindestens einen Index hat, der keinem Datenblock zugeordnet ist, wird als *Datei mit eingesparten Datenblöcken*, *Datei mit freien Bereichen* oder Sparse-Datei bezeichnet. Einer Datei mit freien Bereichen ist eine Größe zugeordnet, aber es sind nicht so viele Datenblöcke zugeordnet, wie für die angegebene Größe erforderlich wäre. Um festzustellen, ob einer Datei weniger Datenblöcke zugeordnet sind, als ihrer Größe entspräche, können Sie den Befehl **fileplace** ausführen. Dieser Befehl zeigt alle Blöcke in der Datei an, die derzeit nicht zugeordnet sind.

Anmerkung: In den meisten Fällen kann auch mit dem Befehl **du** festgestellt werden, ob die Anzahl der Datenblöcke, die einer Datei zugeordnet ist, mit der eigentlich zum Speichern einer Datei dieser Größe erforderlichen Anzahl von Datenblöcken übereinstimmt. Ein komprimiertes Dateisystem kann dasselbe Verhalten für Dateien aufweisen, die keine eingesparten Datenblöcke haben.

Eine Datei mit freien Bereichen wird erstellt, wenn eine Anwendung eine Datei erweitert, indem sie den Dateizeiger außerhalb der derzeit zugeordneten Indizes positioniert, aber die Daten, die geschrieben werden, nicht alle der neu zugeordneten Indizes verbrauchen. Die neue Dateigröße spiegelt den am weitesten entfernten Schreibvorgang in die Datei wider.

Wenn ein Abschnitt einer Datei gelesen wird, der nicht zugeordnete Datenblöcke hat, wird ein Puffer mit Nullen zurückgegeben. Wenn ein Abschnitt einer Datei geschrieben wird, der nicht zugeordnete Datenblöcke hat, werden die erforderlichen Datenblöcke zugeordnet und die Daten geschrieben.

Dieses Verhalten kann sich auf die Befehle für Dateibearbeitung und -archivierung auswirken. Wenn Sie die folgenden Befehle ausführen, werden für die freien Bereiche einer solchen Datei Datenblöcke zugeordnet:

- **cp**
- **mv**
- **tar**
- **cpio**

Anmerkung: Beim Befehl **mv** gilt dies nur für das Verschieben einer Datei in ein anderes Dateisystem. Wenn die Datei innerhalb desselben Dateisystems verschoben wird, bleibt die Datei eine Datei mit freien Bereichen.

Wenn eine Datei mit den zuvor genannten Befehlen kopiert oder wiederhergestellt wird, werden ihr Datenblöcke in einer Anzahl zugeordnet, die ihrer Größe entspricht. Die folgenden Archivierungsbefehle jedoch erhalten die Charakteristika einer Datei mit freien Bereichen bzw. erstellen aktiv eine Datei mit freien Bereichen:

- **backup**
- **restore**
- **pax**

Da es möglich ist, die Ressourcen eines Dateisystems mit Dateien mit freien Bereichen überzubelegen, sollten Dateien dieses Typs mit Sorgfalt verwendet und gepflegt werden.

JFS und große Dateien:

Sie können in einem Dateisystem des Typs JFS große Dateien erstellen.

Alle JFS2-Dateisysteme unterstützen große Dateien.

Dateisysteme, die große Dateien unterstützen, können mit dem Befehl **crfs** und dem Befehl **mkfs** erstellt werden. Beide Befehle haben eine Option (`bf=true`), mit der die Unterstützung für große Dateien in Dateisystemen aktiviert werden kann. Sie können zum Erstellen dieser Dateisysteme auch SMIT verwenden.

In Dateisystemen, die große Dateien unterstützen, werden Dateidaten, die unterhalb eines Datei-Offset von 4 MB gespeichert werden, in 4096-Byte-Blöcken zugeordnet. Dateidaten, die nach dem 4-MB-Datei-Offset gespeichert werden, werden mit großen Plattenblöcken zugeordnet, die eine Größe von 128 KB haben. Diese großen Plattenblöcke sind eigentlich 32 zusammenhängende 4096-Byte-Blöcke.

In einem regulären Dateisystem benötigt eine 132-MB-Datei beispielsweise 33K 4-KB-Plattenblöcke (33 einfach indirekte Blöcke mit jeweils 1024 4-KB-Plattenadressen). Eine 132 MB große Datei in einem Dateisystem, das große Dateien unterstützt, belegt 1024 4-KB-Plattenblöcke und 1024 128-KB-Plattenblöcke. Die Geometrie großer Dateien erfordert nur zwei einzelne einfach indirekte Blöcke für eine Datei mit 132 MB. Große und reguläre Dateitypen erfordern einen doppelt indirekten Block.

Große Plattenblöcke erfordern 32 zusammenhängende 4-KB-Blöcke. Wenn Sie in große Dateien schreiben, die über das 4-MB-Offset hinausgehen, schlägt das Datei-Offset mit ENOSPC fehl, wenn das Dateisystem keine 32 freien zusammenhängenden 4-KB-Blöcke enthält.

Anmerkung: Das Dateisystem kann Tausende freier Blöcke haben. Wenn aber keine 32 davon zusammenhängend sind, scheitert die Zuordnung.

Der Befehl **defragfs** reorganisiert die Plattenblöcke, so dass größere zusammenhängende freie Blockbereiche verfügbar werden.

Das JFS muss alle neuen Plattenzuordnungen initialisieren. Das JFS startet die Kernelprozedur `kproc`, mit der der für Dateien zugeordnete Speicherbereich zunächst mit Nullen aufgefüllt wird, wenn das erste Dateisystem, das große Dateien unterstützt, auf dem System angehängt wird. Die Prozedur `kproc` bleibt geladen, nachdem das Dateisystem mit Unterstützung großer Dateien erfolgreich abgehängt wurde.

Datenkomprimierung in JFS:

JFS unterstützt fragmentierte und komprimierte Dateisysteme, die Plattenspeicherplatz einsparen, indem zugelassen wird, dass ein logischer Block in Einheiten oder "Fragmenten", die kleiner sind als die vollständige Blockgröße von 4096 Bytes, auf der Platte gespeichert werden.

Die Datenverdichtung wird für JFS2 nicht unterstützt.

In einem fragmentierten Dateisystem wird nur der letzte logische Block von Dateien, die kleiner sind als 32 KB, auf diese Weise gespeichert. Die Fragmentunterstützung ist also nur für Dateisysteme von Vorteil, die viele kleine Dateien enthalten. Die Datenverdichtung ermöglicht jedoch, dass alle logischen Blöcke von Dateien beliebiger Größe in einem oder mehreren zusammenhängenden Fragmenten gespeichert werden. Durchschnittlich kann durch Datenkomprimierung eine Einsparung beim Plattenspeicherplatz um den Faktor 2 erzielt werden.

Die Verwendung von Fragmenten und Datenkomprimierung erhöht jedoch das Risiko, dass freier Speicherplatz auf der Platte fragmentiert wird. Fragmente, die einem logischen Block zugeordnet werden, müssen auf der Platte zusammenhängend sein. In einem Dateisystem, in dem freier Speicherplatz fragmentiert wird, kann es schwierig sein, genügend zusammenhängende Fragmente für die Zuordnung eines logischen Blocks zu finden, selbst wenn die Gesamtanzahl freier Fragmente höher ist als der Bedarf für den logischen Block. Das JFS entschärft die Fragmentierung freien Speicherplatzes durch die Bereitstellung des Programms **defragfs**, das ein Dateisystem "defragmentiert", indem es die Menge zusammenhängenden freien Speicherplatzes erhöht. Dieses Dienstprogramm kann für fragmentierte und kompri-

mierte Dateisysteme verwendet werden. Die durch Fragmente und Datenkomprimierung erzielten Einsparungen an Plattenspeicherplatz können beträchtlich sein, wobei das Problem der Fragmentierung freien Speicherplatzes kontrollierbar bleibt.

Die Datenkomprimierung im aktuellen JFS ist mit früheren Versionen dieses Betriebssystems kompatibel. Die API, die alle Systemaufrufe enthält, ist in beiden JFS-Versionen dieselbe.

Zugehörige Konzepte:

„Segmentierung des Plattenspeicherplatzes in JFS und JFS2“ auf Seite 490

Viele UNIX-Dateisysteme ordnen den logischen Blöcken, die für die logische Unterteilung von Dateien und Verzeichnissen verwendet werden, nur zusammenhängenden Speicherplatz in Einheiten identischer Größe zu. Diese Zuordnungseinheiten werden gewöhnlich als *Plattenblöcke* bezeichnet. Ein Plattenblock wird ausschließlich dazu verwendet, um die Daten, die in einem logischen Block einer Datei oder eines Verzeichnisses enthalten sind, zu speichern.

Implementierung der JFS-Datenkomprimierung:

Datenkomprimierung ist ein Attribut eines Dateisystems, das angegeben wird, wenn das Dateisystem mit dem Befehl **crfs** oder **mkfs** erstellt wird. Sie können SMIT verwenden, um die Datenkomprimierung anzugeben.

Achtung: Das Stammdateisystem (/) darf nicht komprimiert werden. Die Komprimierung des Dateisystems /usr wird nicht empfohlen, da **installp** in der Lage sein muss, die Größe des Dateisystems für Aktualisierungen und Neuinstallationen genau zu berechnen.

Die Komprimierung gilt nur für reguläre Dateien und lange symbolische Verbindungen in solchen Dateisystemen. Die Fragmentunterstützung gilt weiterhin für Verzeichnisse und Metadaten, die nicht komprimiert sind. Jeder logische Block einer Datei wird einzeln komprimiert, bevor er auf die Platte geschrieben wird. Eine solche Komprimierung vereinfacht die wahlfreien Zugriffe und Aktualisierungen, während im Vergleich zur Komprimierung von Daten in größeren Einheiten nur ein kleiner Teil des freigegebenen Plattenspeicherplatzes verloren geht.

Nach der Komprimierung belegt ein logischer Block gewöhnlich weniger als 4096 Bytes Plattenspeicherplatz. Der komprimierte logische Block wird auf die Platte geschrieben, und es wird nur die erforderliche Anzahl benachbarter Fragmente für die Speicherung des Blocks zugeordnet. Wenn ein logischer Block nicht komprimiert wird, wird er in nicht komprimierter Form auf die Platte geschrieben, und es werden 4096 Bytes benachbarter Fragmente zugeordnet.

Der Befehl **lsfs -q** zeigt den aktuellen Wert für die Komprimierung an. Sie können SMIT auch verwenden, um die Datenkomprimierung zu identifizieren.

Zugehörige Konzepte:

„Implizites Verhalten der JFS-Datenkomprimierung“

Da ein Programm, das eine Datei schreibt, keine ENOSPC-Bedingung (Out-of-Space, kein Speicherplatz mehr vorhanden) nach einer erfolgreichen Schreiboperation (oder einer erfolgreichen Speicherung zugeordneter Dateien) erwartet, muss sichergestellt werden, dass ausreichend Speicherplatz verfügbar ist, wenn logische Blöcke auf die Platte geschrieben werden.

Implizites Verhalten der JFS-Datenkomprimierung:

Da ein Programm, das eine Datei schreibt, keine ENOSPC-Bedingung (Out-of-Space, kein Speicherplatz mehr vorhanden) nach einer erfolgreichen Schreiboperation (oder einer erfolgreichen Speicherung zugeordneter Dateien) erwartet, muss sichergestellt werden, dass ausreichend Speicherplatz verfügbar ist, wenn logische Blöcke auf die Platte geschrieben werden.

Dies wird erreicht, indem einem logischen Block 4096 Bytes zugeordnet werden, wenn er zum ersten Mal geändert wird, so dass selbst dann Plattenspeicherplatz verfügbar ist, wenn der Block nicht komprimiert

wird. Wenn keine 4096-Byte-Zuordnung verfügbar ist, gibt das System eine ENOSPC- oder EDQUOT-Fehlerbedingung zurück, obwohl unter Umständen genügend Plattenspeicherplatz verfügbar ist, um den komprimierten logischen Block unterzubringen. Eine ENOSPC-Bedingung wird wahrscheinlich frühzeitig ausgegeben, wenn die Grenzwerte für die Datenträgerkontingente oder die Kapazität des Dateisystems nahezu ausgeschöpft sind.

Komprimierte Dateisysteme können außerdem das folgende Verhalten aufweisen:

- Da einem Block zunächst 4096 Bytes zugeordnet werden, wird bei bestimmten Systemaufrufen unter Umständen ein ENOSPC- oder EDQUOT-Fehler ausgelöst. Möglicherweise wurde eine alte Datei mit dem Systemaufruf **mmap** zugeordnet, und das Speichern an einer zuvor beschriebenen Position kann zu einem ENOSPC-Fehler führen.
- Mit Datenkomprimierung bleibt ein vollständiger Plattenblock einem geänderten Block so lange zugeordnet, bis er auf die Platte geschrieben wird. Wenn der Block eine zuvor festgeschriebene Zuordnung von weniger als einem vollständigen Block hatte, ist der vom Block gebundene Plattenspeicherplatz die Summe der beiden Zuordnungen. Die vorherige Zuordnung wird erst dann aufgehoben, wenn die Datei (I-Node) festgeschrieben ist. Dies ist der Fall bei normalen Fragmenten. Die Anzahl der logischen Blöcke in einer Datei, die zuvor festgeschriebene Zuordnungen haben kann, ist für normale Fragmente maximal 1, kann aber mit Komprimierung so hoch wie die Anzahl der Blöcke in einer Datei sein.
- Die zuvor festgeschriebenen Ressourcen für einen logischen Block werden erst dann freigegeben, wenn der Systemaufruf **fsync** oder **sync** vom Anwendungsprogramm ausgeführt wird.
- Der Systemaufruf **stat** gibt die Anzahl der Fragmente an, die einer Datei zugeordnet sind. Die zurückgemeldete Anzahl basiert auf den 4096 Bytes, die geänderten, aber noch nicht geschriebenen Blöcken zugeordnet werden, und der komprimierten Größe nicht geänderter Blöcke. Zuvor festgeschriebene Ressourcen werden vom Systemaufruf **stat** nicht mitgezählt. Der Systemaufruf **stat** meldet die korrekte Anzahl zugeordneter Fragmente nach einer I-Node-Festschreibungsoperation zurück, wenn keiner der geänderten Blöcke komprimiert wurde. Für die aktuelle Zuordnung werden Datenträgerkontingente berechnet. Da die logischen Blöcke einer Datei auf die Platte geschrieben werden, verringert sich die Anzahl der ihnen zugeordneten Fragmente nach der Komprimierung. Damit ändern sich auch die Datenträgerkontingente und das Ergebnis von **stat**.

Zugehörige Konzepte:

„Implementierung der JFS-Datenkomprimierung“ auf Seite 498

Datenkomprimierung ist ein Attribut eines Dateisystems, das angegeben wird, wenn das Dateisystem mit dem Befehl **crfs** oder **mkfs** erstellt wird. Sie können SMIT verwenden, um die Datenkomprimierung anzugeben.

Algorithmus für JFS-Datenkomprimierung:

Der Komprimierungsalgorithmus ist eine IBM Version von LZ. Im Allgemeinen komprimieren LZ-Algorithmen Daten, indem sie das zweite oder spätere Vorkommen einer bestimmten Zeichenfolge durch einen Zeiger darstellen, der die Position des ersten Vorkommens der Zeichenfolge mitsamt Länge angibt.

Zu Beginn des Komprimierungsprozesses wurden noch keine Zeichenfolgen identifiziert. Deshalb muss zumindest das erste Datenbyte unverändert dargestellt werden, das 9 Bits (0,Byte) erfordert. Nachdem ein bestimmtes Datenvolumen komprimiert wurde, z. B. N Bytes, sucht der Kompressor nach der längsten Zeichenfolge in den N Bytes, die der Zeichenfolge entspricht, die beim nächsten nicht verarbeiteten Byte beginnt. Wenn die längste Übereinstimmung die Länge 0 oder 1 hat, wird das nächste Byte unverändert codiert. Andernfalls wird die Zeichenfolge als Zeiger/Länge-Paar dargestellt, und die Anzahl der verarbeiteten Bytes wird um die Länge erhöht. Architektonisch unterstützt die IBM Version von LZ für N die Werte 512, 1024 und 2048. IBM LZ erlaubt die Codierung von Zeiger/Länge-Paaren und unveränderten Zeichen. Der Zeiger ist ein Feld fester Länge mit der Größe $\log_2 N$, während die Länge als Feld variabler Länge codiert ist.

Leistungseinbußen bei der JFS-Datenkomprimierung:

Da die Datenkomprimierung eine Erweiterung der Fragmentunterstützung ist, gilt die Leistung bezüglich der Fragmente auch für die Datenkomprimierung.

Komprimierte Dateisysteme wirken sich folgendermaßen auf die Leistung aus:

- Die Komprimierung und Dekomprimierung von Daten kann eine erhebliche Zeit in Anspruch nehmen, sodass die Verwendbarkeit eines komprimierten Dateisystems für einige Benutzerumgebungen eingeschränkt sein kann.
- Die meisten regulären UNIX-Dateien werden nur einmal geschrieben, aber einige werden an Ort und Stelle aktualisiert. Für letztere Dateien bringt die Datenkomprimierung weitere Leistungseinbußen mit sich, weil 4096 Bytes Plattenspeicherplatz zugeordnet werden müssen, wenn ein logischer Block geändert wird, und anschließend erneut Plattenspeicherplatz zugeordnet werden muss, wenn der logische Block auf die Platte geschrieben wird. Diese zusätzliche Zuordnungsaktivität ist für reguläre Dateien in einem nicht komprimierten Dateisystem nicht erforderlich.
- Die Datenkomprimierung erhöht die Anzahl der Prozessorzyklen. Für den Softwarekompressor beläuft sich die Anzahl der Komprimierungszyklen auf ungefähr 50 Zyklen pro Byte und für die Dekomprimierung auf 10 Zyklen pro Byte.

JFS-Onlinesicherungen und JFS2-Momentaufnahmen:

Sie können ein zeitpunktgesteuertes Image eines JFS-Dateisystems oder eines JFS2-Dateisystems erstellen, das Sie später zu Sicherungszwecken verwenden können. Anforderungen und Verhalten dieses Image sind je nach Dateisystemtyp verschieden.

Für ein JFS-Dateisystem können Sie eine schreibgeschützte statische Kopie einer gespiegelten Kopie des Dateisystems abtrennen. Gewöhnlich wird eine gespiegelte Kopie aktualisiert, wenn das ursprüngliche Dateisystem aktualisiert wird. Diese Zeitpunktkopie ändert sich jedoch nicht. Sie bleibt ein stabiles Image des Dateisystems zu dem Zeitpunkt, zu dem die Kopie erstellt wurde. Wenn dieses Image für die Sicherung verwendet wird, sind Änderungen, die nach dem Beginn der Prozedur zum Erstellen des Image vorgenommen wurden, möglicherweise nicht in der Sicherungskopie enthalten. Deshalb wird empfohlen, alle Aktivitäten im Dateisystem auf ein Minimum zu reduzieren, während die Abtrennung vollzogen wird. Alle Änderungen, die nach dem Abtrennen vorgenommen werden, sind nicht in der Sicherungskopie enthalten.

Bei einem JFS2-Dateisystem wird das zeitpunktgesteuerte Image als *Momentaufnahme* bezeichnet. Die Momentaufnahme bleibt statisch und behält dieselben Sicherheitseinstellungen, die das ursprüngliche Dateisystem (das so genannte *snappedFS*) beim Erstellen der Momentaufnahme hatte. Sie können eine JFS2-Momentaufnahme auch ohne Abhängen oder Stilllegung des Dateisystems erstellen. Eine JFS2-Momentaufnahme kann als Onlinesicherung des Dateisystems verwendet werden, um auf die Dateien und Verzeichnisse zuzugreifen, wie sie bei der Erstellung der Momentaufnahme vorhanden waren, oder um die Sicherung auf einem austauschbaren Datenträger zu speichern. Folgendes ist bei JFS2-Momentaufnahmen zu beachten:

- Ein Image mit einer Momentaufnahme des Stammverzeichnisses (/) oder des Dateisystems /usr wird überschrieben, wenn ein Warmstart des Systems durchgeführt wird. Momentaufnahmen anderer Dateisysteme können bewahrt werden, indem das Dateisystem vor dem Warmstart abgehängt wird. Momentaufnahmen in AIX 5.2 with 5200-01 sind wiederherstellbar. Wenn der Befehl **fsck** oder **logredo** für ein JFS2-Dateisystem mit einer unter AIX 5.2 with 5200-01 erstellten Momentaufnahme ausgeführt wird, bleibt die Momentaufnahme erhalten. Ein ordnungsgemäß abgehängtes Dateisystem mit einer unter AIX 5.2 erstellten Momentaufnahme ist auch wiederherstellbar, sobald es an ein System mit AIX 5.2 with 5200-01 angehängt wird.
- Die Ausführung des Befehls **defragfs** für ein Dateisystem mit Momentaufnahmen wird nicht empfohlen. Jeder Block, der während der Defragmentierung verschoben wird, muss auch in die Momentaufnahme kopiert werden. Dieser Vorgang ist zeitaufwändig und eine Verschwendung von Plattenspeicherplatz auf dem logischen Datenträger mit der Momentaufnahme.

- Wenn der Speicherplatz für eine Momentaufnahme knapp wird, werden alle Momentaufnahmen für das `snappedFS` gelöscht. Beim Auftreten dieses Fehlers wird ein Eintrag in das Fehlerprotokoll geschrieben.
- Wenn das Schreiben in eine Momentaufnahme scheitert, werden alle Momentaufnahmen für dieses `snappedFS` gelöscht. Beim Auftreten dieses Fehlers wird ein Eintrag in das Fehlerprotokoll geschrieben.
- Eine Momentaufnahme, die auf einem System mit AIX 5.2 with 5200-01 erstellt oder aufgerufen wird, kann auf einem System mit AIX 5.2 nicht aufgerufen werden. Diese Momentaufnahmen müssen gelöscht werden, damit das Dateisystem angehängt werden kann.
- Ein JFS2-Dateisystem, das eine in AIX 5.3 erstellte Momentaufnahme enthält, kann in den Releases vor AIX 5.2 with 5200-01 nicht geöffnet werden. Wenn das System wieder zurückverschoben werden soll, müssen die Momentaufnahmen zuerst gelöscht werden, damit der Zugriff auf das Dateisystem zugelassen wird.

JFS-Onlinesicherung:

Sie können ein zeitpunktgesteuertes Image eines JFS-Dateisystems erstellen, das Sie anschließend für Sicherungen verwenden können.

Für ein JFS-Dateisystem können Sie eine schreibgeschützte statische Kopie einer gespiegelten Kopie des Dateisystems abtrennen. Gewöhnlich wird eine gespiegelte Kopie aktualisiert, wenn das ursprüngliche Dateisystem aktualisiert wird. Diese Zeitpunktkopie ändert sich jedoch nicht. Sie bleibt ein stabiles Image des Dateisystems zu dem Zeitpunkt, zu dem die Kopie erstellt wurde. Wenn dieses Image für die Sicherung verwendet wird, sind Änderungen, die nach dem Beginn der Prozedur zum Erstellen des Image vorgenommen wurden, möglicherweise nicht in der Sicherungskopie enthalten. Deshalb wird empfohlen, alle Aktivitäten im Dateisystem auf ein Minimum zu reduzieren, während die Abtrennung vollzogen wird. Alle Änderungen, die nach dem Abtrennen vorgenommen werden, sind nicht in der Sicherungskopie enthalten.

JFS2-Momentaufnahmen:

Sie können ein zeitpunktgesteuertes Image eines JFS2-Dateisystems erstellen, das Sie anschließend für Sicherungen verwenden können.

Bei einem JFS2-Dateisystem wird das zeitpunktgesteuerte Image *Momentaufnahme* genannt. Die Momentaufnahme bleibt statisch und behält dieselben Sicherheitseinstellungen, die das ursprüngliche Dateisystem (das so genannte *snappedFS*) beim Erstellen der Momentaufnahme hatte. Sie können eine JFS2-Momentaufnahme auch ohne Abhängen oder Stilllegung des Dateisystems erstellen. Eine JFS2-Momentaufnahme kann für folgende Zwecke verwendet werden:

- Für den Zugriff auf Dateien oder Verzeichnisse mit dem Stand, den sie beim Erstellen der Momentaufnahme hatten.
- Für die Sicherung austauschbarer Datenträger.

Es gibt zwei Arten von JFS2-Momentaufnahmen: interne und externe. Eine externe JFS2-Momentaufnahme wird in einem vom Dateisystem getrennten logischen Datenträger erstellt. Die externe Momentaufnahme kann getrennt vom Dateisystem an einem eigenen eindeutigen Mountpunkt angehängt werden.

Eine interne JFS2-Momentaufnahme wird in demselben logischen Datenträger erstellt, in dem sich das Dateisystem befindet, und ordnet Blöcke aus dem Dateisystem zu. Eine interne Momentaufnahme kann aus dem unsichtbaren Verzeichnis `.snapshot` im Stammverzeichnis des JFS2-Dateisystems mit der Momentaufnahme aufgerufen werden. Ein JFS2-Dateisystem muss bei seiner Erstellung für die Unterstützung von internen Momentaufnahmen aktiviert worden sein.

JFS2-Momentaufnahmen unterstützen keine Überprüfung von Dateisystemkontingenten. Der Befehl **repquota** kann nicht für die Momentaufnahme verwendet werden, um den Status des Kontingents zu er-

mitteln. Die zeitpunktgesteuerten Kontingentdaten werden beibehalten, wenn Sie das Dateisystemimage auf das Image der Momentaufnahme zurücksetzen. Beachten Sie folgende Hinweise, die für externe und interne JFS2-Momentaufnahmen gelten:

- Eine externe Momentaufnahme, die auf einem System mit AIX 5.2 with 5200-01 erstellt oder aufgerufen wird, kann auf einem System mit AIX 5.2 nicht aufgerufen werden. Diese Momentaufnahmen müssen gelöscht werden, damit das Dateisystem angehängt werden kann.
- Ein JFS2-Dateisystem, das eine in AIX 5.3 erstellte Momentaufnahme enthält, kann in den Releases vor AIX 5.2 with 5200-01 nicht geöffnet werden. Wenn das System wieder zurückverschoben werden soll, müssen die Momentaufnahmen zuerst gelöscht werden, damit der Zugriff auf das Dateisystem zugelassen wird.
- Es wird empfohlen, den Befehl **defragfs** nicht für ein JFS2-Dateisystem mit externen Momentaufnahmen auszuführen, da jeder Block, der während der Defragmentierung verschoben wird, auch in die Momentaufnahme kopiert werden muss. Dieser Vorgang ist zeitaufwändig und eine Verschwendung von Plattenspeicherplatz auf dem logischen Datenträger mit der Momentaufnahme.
- Wenn der Speicherplatz für eine Momentaufnahme zu knapp wird oder wenn eine externe Momentaufnahme scheitert, werden alle externen Momentaufnahmen für das snappedFS als ungültig gekennzeichnet. Jeder weitere Zugriff auf die Momentaufnahme wird scheitern. Diese Fehler haben einen Eintrag im Fehlerprotokoll zur Folge.

Hinweise für interne JFS2-Momentaufnahmen:

- Interne Momentaufnahmen werden beibehalten, wenn der Befehl **logredo** auf einem JFS2-Dateisystem mit einer internen Momentaufnahme ausgeführt wird.
- Interne Momentaufnahmen werden entfernt, wenn der Befehl **fsck** ein JFS2-Dateisystem ändern muss, um es zu reparieren.
- Wenn der Speicherplatz für eine interne Momentaufnahme zu knapp wird oder wenn das Schreiben in eine interne Momentaufnahme scheitert, werden alle internen Momentaufnahmen für das snappedFS als ungültig gekennzeichnet. Jeder weitere Zugriff auf die internen Momentaufnahmen wird scheitern. Diese Fehler haben einen Eintrag im Fehlerprotokoll zur Folge.
- Interne Momentaufnahmen können nicht gesondert angehängt werden. Sie können interne Momentaufnahmen sofort nach ihrer Erstellung im Verzeichnis `.snapshot` des Stammverzeichnis des Dateisystems aufrufen. Folglich können Sie die internen Momentaufnahmen über einen NFS-Server aufrufen, ohne dass es erforderlich wäre, einen separaten Mountpunkt für die Momentaufnahme zu exportieren.
- Interne Momentaufnahmen sind nicht kompatibel mit AIX-Releases vor AIX 6.1. Ein JFS2-Dateisystem, das zur Unterstützung interner Momentaufnahmen erstellt wurde, kann auf einem Release vor AIX nicht geändert werden.
- Ein JFS2-Dateisystem, das zur Unterstützung interner Momentaufnahmen erstellt wurde, kann auch Extended Attributes Version 2 unterstützen.
- Ein JFS2-Dateisystem mit internen Momentaufnahmen kann nicht mit einer DMAPI (Data Management Application Programming Interface, Anwendungsprogrammierschnittstelle für Datenmanagement) verwendet werden.
- Der Befehl **defragfs** kann nicht für ein JFS2-Dateisystem mit internen Momentaufnahmen verwendet werden.
- Das Verzeichnis `.snapshot` wird vom Systemaufruf **readdir()** nicht zurückgegeben. Dadurch wird der unbeabsichtigte Zugriff auf die Momentaufnahmen verhindert. Alle Systemaufrufe oder Befehle, die vom Systemaufruf **readdir()** abhängen, scheitern für das Verzeichnis `.snapshot` (beispielsweise können der Befehl **/bin/pwd** und der Systemaufruf **getcwd()** des Verzeichnisses `.snapshot` das übergeordnete Verzeichnis nicht finden).

Kompatibilität und Migration:

JFS-Dateisysteme sind in AIX 5.1 und AIX 5.2 vollständig kompatibel. Frühere unterstützte Versionen des Betriebssystems sind mit dem aktuellen JFS kompatibel, obwohl Dateisysteme mit einer Fragmentgröße,

einem NBPI-Wert oder eine Zuordnungsgruppengröße, die bzw. der nicht dem Standardwert entspricht, besonderer Aufmerksamkeit bedürfen, wenn sie auf eine frühere Version migriert werden.

Anmerkung: JFS-Dateisysteme werden auf Platten mit einer Sektorengröße von 4 KB nicht unterstützt. Wenn Sie ein Dateisystem erstellen oder Sicherungsoperationen durchführen, müssen Sie deshalb sicherstellen, dass die Platten keine Sektorengröße von 4 KB haben.

JFS2-Dateisysteme sind mit Ausnahme von Momentaufnahmen in AIX 5.1 und AIX 5.2, aber nicht mit früheren Versionen des Betriebssystems kompatibel. JFS2-Dateisysteme mit Momentaufnahmen werden in AIX 5.1 nicht unterstützt. Stellen Sie sicher, dass alle JFS2-Dateisysteme ordnungsgemäß abgehängt werden, bevor Sie auf eine frühere Version von AIX umstellen, weil der Befehl **logredo** unter Umständen nicht für Dateisysteme ausgeführt werden kann, die für ein höheres Release erstellt wurden.

Anmerkung: Der Zugriff auf JFS2-Dateisysteme, die für das Format der Version 2 erstellt bzw. in dieses Format konvertiert wurden, ist in früheren Releases von AIX nicht möglich.

In der folgenden Liste sind die Aspekte beschrieben, die zu Problemen mit Dateisystemen führen können, die in früheren Versionen des Betriebssystems erstellt wurden:

JFS-Dateisystemimages

Jedes JFS-Dateisystemimage, das mit der Standardfragmentgröße, einem NBPI-Wert von 4096 Bytes und der Standardzuordnungsgruppengröße (agsize) 8 erstellt wird, kann durch ein JFS-Dateisystemimage ausgetauscht werden, das in AIX 4.3 und höheren Versionen dieses Betriebssystems erstellt wird, ohne dass spezielle Migrationsaktivitäten ausgeführt werden müssen.

Anmerkung: JFS2-Momentaufnahmen: JFS2-Momentaufnahmen, die in AIX 5L Version 5.2 mit 5200-01 Recommended Maintenance Package erstellt oder aufgerufen werden, sind in früheren Releases nicht zugänglich. Diese Momentaufnahmen müssen gelöscht werden, damit das Dateisystem angehängt werden kann.

Sicherung und Wiederherstellung bei JFS-Dateisystemen

Die Sicherung und anschließende Wiederherstellung zwischen JFS-Dateisystemen mit unterschiedlichen Blockgrößen sind zwar möglich, aber aufgrund der höheren Plattenauslastung können Wiederherstellungsoperationen scheitern, weil nicht genügend freie Blöcke verfügbar sind, wenn die Blockgröße des Quelldateisystems kleiner ist als die Blockgröße des Zieldateisystems. Dies ist von besonderem Interesse bei vollständigen Dateisystemsicherungen und -wiederherstellungen und kann selbst dann auftreten, wenn die Gesamtgröße des Zieldateisystems höher ist als die des Quelldateisystems.

Obwohl eine Sicherung eines komprimierten Dateisystems mit anschließender Wiederherstellung in einem nicht komprimierten Dateisystem und zwischen komprimierten Dateisystemen mit unterschiedlichen Fragmentgrößen möglich ist, können Wiederherstellungsoperationen aufgrund der erweiterten Plattenauslastung komprimierter Dateisysteme scheitern, wenn nicht genügend Plattenspeicherplatz verfügbar ist. Dies ist von besonderem Interesse bei vollständigen Dateisystemsicherungen und -wiederherstellungen und kann selbst dann auftreten, wenn die Gesamtgröße des Zieldateisystems höher ist als die des Quelldateisystems.

Einschränkungen für JFS- und JFS2-Einheitentreiber

Außerdem müssen Einheitentreiber eine Adressierbarkeit von Plattenblöcken unterstützen, die kleiner-gleich der JFS-Fragmentgröße bzw. der JFS2-Blockgröße des Dateisystems ist. Wenn ein JFS-Dateisystem beispielsweise mit einem vom Benutzer bereitgestellten Einheitentreiber für RAM-Disks erstellt wurde, muss der Treiber 512-Byte-Blöcke zulassen, um ein Dateisystem aufzunehmen, das 512-Byte-Fragmente enthält. Wenn der Treiber nur die Adressierung auf Seitenebene zulässt, kann nur ein JFS mit einer Fragmentgröße von 4096 Bytes verwendet werden.

Ein JFS auf einen anderen physischen Datenträger kopieren:

Sie können ein JFS-Dateisystem unter Bewahrung der Dateisystemintegrität auf einen anderen physischen Datenträger kopieren.

Im Folgenden wird beschrieben, wie Sie ein JFS-Dateisystem unter Bewahrung der Dateisystemintegrität auf einen anderen physischen Datenträger kopieren.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

Gehen Sie wie folgt vor, um ein JFS unter Bewahrung der Dateisystemintegrität auf einen anderen physischen Datenträger zu kopieren:

1. Stoppen Sie alle Aktivitäten für das Dateisystem, das Sie kopieren möchten. Sie können sich nur dann auf den Inhalt der Sicherung verlassen, wenn die Anwendung, die das Dateisystem verwendet, keine Aktionen durchführt und sich das Dateisystem in einem bekannten Status befindet.

2. Spiegeln Sie den logischen Datenträger. Geben Sie hierfür in der Befehlszeile den folgenden SMIT-Direktaufruf ein:

```
smit mklvcopy
```

3. Kopieren Sie das Dateisystem mit dem folgenden Befehl:

```
chfs -a splitcopy=/backup -a copy=2 /testfs
```

Der Parameter **splitcopy** des Flags **-a** weist den Befehl an, eine gespiegelte Kopie des Dateisystems abzutrennen und dieses im Lesezugriff an den neuen Mountpunkt anzuhängen. Auf diese Weise erhalten Sie eine Kopie des Dateisystems mit konsistenten aufgezeichneten Metadaten, die für Sicherungen verwendet werden kann.

4. Wenn Sie die gespiegelte Kopie auf einen anderen Mountpunkt kopieren möchten, verwenden Sie den folgenden SMIT-Direktpfad:

```
smit cplv
```

Damit ist die Kopie des Dateisystems nutzbar.

Zugehörige Informationen:

mklv command

CD-ROM-Dateisystem und UDF-Dateisystem:

Das CD-ROM-Dateisystem (CDRFS, CD-ROM File System) ist eine schreibgeschützte lokale Dateisystemimplementierung, die auf CD-ROM-Medien, CD-RW-Medien (schreibgeschützt) und DVD-ROM-Medien gespeichert werden kann. Die maximale CDRFS-Dateigröße sind unabhängig vom verwendeten Medium 2 GB. Das UDF-Dateisystem (Universal Disk Format) ist eine beschreibbare lokale Dateisystemimplementierung, die schreibgeschützt auf DVD-ROM-Medien oder mit Lese/Schreibzugriff auf DVD-RAM-Medien gespeichert werden kann.

CDs werden automatisch angehängt, aber diese Funktion kann inaktiviert werden. Wenn die Funktion inaktiviert ist, können Sie das CDRFS mit dem Befehl **cdmount** anhängen.

AIX unterstützt die folgenden CDRFS-Datenträger- und -Dateistrukturformate:

| Typ | Beschreibung |
|--|---|
| Standard ISO 9660:1988(E) | Das CDRFS unterstützt Stufe 3 des Standards ISO 9660 für den Datenaustausch und Stufe 1 der Implementierung. |
| High Sierra Group Specification | Dieser Vorgänger des Standards ISO 9660 unterstützt die Abwärtskompatibilität mit früheren CD-ROMs. |
| Rock Ridge Group Protocol | Definiert Erweiterungen für den Standard ISO 9660, die vollständig kompatibel mit diesem sind und die vollständige Semantik für POSIX-Dateisysteme unterstützen, die auf dem System Use Sharing Protocol (SUSP) und dem Rock Ridge Interchange Protocol (RRIP) basiert. Das Anhängen von und Zugreifen auf die CD-ROM erfolgt wie bei jedem anderen UNIX-Dateisystem. |
| CD-ROM eXtended Architecture File Format (nur im Sektorformat Mode 2 Form 1) | Das Dateiformat CD-ROM eXtended Architecture definiert Erweiterungen für den Standard ISO 9660, die in CD-ROM-basierenden Multimediaanwendungen, wie z. B. Photo CD, verwendet werden. |

Für alle Datenträger- und Dateistrukturformate gelten die folgenden Einschränkungen:

- nur Datenträgergruppen mit einem Datenträger
- nur nicht verzahnte Dateien

Das CDRFS ist davon abhängig, dass der zugrunde liegende CD-ROM-Einheitentreiber für die Transparenz des physischen Sektorformats (CD-ROM Mode 1 und CD-ROM XA Mode 2 Form 1) und des Mehrfachsatzungsformats der Platten (Zuordnung der Datenträgerdeskriptorgruppe aus dem Datenträgererkennungsbereich der letzten Sitzung) sorgt.

Anmerkung: Das CDRFS muss vom System abgehängt werden, damit Sie die CD-ROM entnehmen können.

Es gibt einen weiteren unterstützten Dateisystemtyp mit dem Namen UDFS. UDFS ist ein schreibgeschütztes Dateisystem, das auf DVD-ROMs gespeichert wird. UDFS muss vom System abgehängt werden, damit Sie den Datenträger entnehmen können. AIX unterstützt das UDFS-Format in den Versionen 1.50, 2.00 und 2.01.

Ein UDFS muss mit NFS im Lesezugriffsmodus exportiert werden. Das Schreiben in ein über NFS angehängtes UDFS wird nicht unterstützt.

Wenn Sie den Befehl **cdmount** verwenden möchten, um ein UDFS mit Lese-/Schreibzugriff automatisch anzuhängen, müssen Sie die Datei `cdromd.conf` editieren. Sie können ein UDFS mit Lese-/Schreibzugriff auch manuell mit dem Befehl **mount** anhängen.

Verzeichnisse

Ein *Verzeichnis* ist eine bestimmte Art von Datei, die nur die Informationen enthält, die für den Zugriff auf Dateien und andere Verzeichnisse erforderlich sind. Folglich beansprucht ein Verzeichnis weniger Speicherplatz als andere Dateiarten.

Dateisysteme bestehen aus Gruppen von Verzeichnissen und den Dateien in diesen Verzeichnissen. Dateisysteme werden in der Regel als umgekehrter Baum dargestellt. Das Stammverzeichnis, dargestellt durch den Schrägstrich (/), definiert ein Dateisystem und erscheint oben im Diagramm des Dateisystembaums.

Im Baumdiagramm verzweigen Verzeichnisse vom Stammverzeichnis ausgehend nach unten und können Dateien und Unterverzeichnisse enthalten. Die Verzweigungen bilden eindeutige Pfade in der Verzeichnisstruktur zu jedem Objekt im Dateisystem.

Sammlungen von Dateien werden in Verzeichnissen gespeichert. Diese Dateisammlungen beziehen sich häufig aufeinander. Durch das Speichern in einer Verzeichnisstruktur kann eine übersichtliche Organisation beibehalten werden.

Eine *Datei* ist eine Sammlung von Daten, die gelesen und in die geschrieben werden kann. Eine Datei kann ein Programm sein, das Sie erstellen, Text, den Sie schreiben, Daten, die Sie erfassen, oder eine Einheit, die Sie verwenden. Befehle, Drucker, Terminals, Korrespondenz und Anwendungsprogramme werden alle in Dateien gespeichert. Dieses Verfahren ermöglicht den Benutzern den einheitlichen Zugriff auf verschiedene Elemente des Systems und verleiht dem Dateisystem eine große Flexibilität.

Mit Verzeichnissen können Sie Dateien und weitere Verzeichnisse gruppieren, um das Dateisystem modular zu strukturieren. Dies verleiht der Dateisystemstruktur Flexibilität und Tiefe.

Verzeichnisse enthalten Verzeichniseinträge. Jeder Eintrag enthält den Namen einer Datei oder eines Unterverzeichnisses sowie eine I-Node-Referenznummer (*I-Node-Nummer*). Zur Verbesserung der Zugriffszeiten und Nutzung des Speicherplatzes werden die Daten in einer Datei an verschiedenen Positionen im Speicher des Computers abgelegt. Die I-Node-Nummer enthält die Adressen, mit denen die verstreuten Datenblöcke, die der Datei zugeordnet sind, wiedergefunden werden. Außerdem werden in der I-Node-Nummer weitere Informationen zur Datei, wie z. B. das letzte Änderungsdatum und das letzte Zugriffsdatum, Zugriffsmodi, Anzahl der Verbindungen, Dateieigner und Dateityp, aufgezeichnet.

Verzeichnisse werden mit einer speziellen Gruppe von Befehlen gesteuert. Beispielsweise können Sie mehrere Namen für eine Datei an dieselbe I-Node-Nummer binden, indem Sie mit dem Befehl **In** Verzeichniseinträge erstellen.

Da Verzeichnisse häufig Informationen enthalten, die nicht allen Benutzern im System zugänglich sein sollten, kann der Verzeichniszugriff geschützt werden. Durch Festlegen von Berechtigungen für ein Verzeichnis können Sie steuern, wer auf das Verzeichnis zugreifen darf und welche Benutzer (sofern überhaupt) Informationen im Verzeichnis ändern dürfen.

Zugehörige Konzepte:

„Zugriffsmodi für Dateien und Verzeichnisse“ auf Seite 314

Jede Datei hat einen Eigner. Wenn eine neue Datei erstellt wird, ist der Benutzer, der die Datei erstellt, der Eigner der Datei. Der Eigner legt einen *Zugriffsmodus* für die Datei fest. Mit den Zugriffsmodi können anderen Systembenutzern die Berechtigungen zum Lesen, Ändern oder Ausführen der Datei erteilt werden. Der Zugriffsmodus einer Datei kann nur vom Dateieigner oder von Benutzern mit Rootberechtigung geändert werden.

Verzeichnistypen:

Verzeichnisse können vom Betriebssystem, vom Systemadministrator und von Benutzern definiert werden.

Die vom System definierten Verzeichnisse enthalten spezielle Systemdateien, wie z. B. Befehle. Ganz oben in der Hierarchie des Dateisystems befindet sich das systemdefinierte Stammverzeichnis `/`. Das Stammverzeichnis `/` enthält im Allgemeinen die folgenden systembezogenen Standardverzeichnisse:

| Eintrag | Beschreibung |
|-----------------------|--|
| <code>/dev</code> | Enthält Gerätedateien für E/A-Einheiten. |
| <code>/etc</code> | Enthält Dateien für die Systeminitialisierung und die Systemverwaltung. |
| <code>/home</code> | Enthält Anmeldeverzeichnisse für Systembenutzer. |
| <code>/tmp</code> | Enthält Dateien, die temporär sind und automatisch nach einer bestimmten Anzahl von Tagen gelöscht werden. |
| <code>/usr</code> | Enthält die Verzeichnisse <code>lpp</code> , <code>include</code> und weitere Systemverzeichnisse. |
| <code>/usr/bin</code> | Enthält ausführbare Benutzerprogramme. |

Einige Verzeichnisse, wie z. B. das Anmelde- oder Ausgangsverzeichnis (`$HOME`), werden vom Systemadministrator definiert und angepasst. Nach der Anmeldung am Betriebssystem ist das Anmeldeverzeichnis das aktuelle Verzeichnis.

Verzeichnisse, die vom Benutzer erstellt werden, heißen *benutzerdefinierte* Verzeichnisse. In diesen Verzeichnissen können Sie Ihre Dateien organisieren und verwalten.

Verzeichnisorganisation:

Verzeichnisse enthalten Dateien, Unterverzeichnisse oder beides. Ein *Unterverzeichnis* ist ein Verzeichnis innerhalb eines Verzeichnisses. Das Verzeichnis mit dem Unterverzeichnis wird als *übergeordnetes Verzeichnis* bezeichnet.

Jedes Verzeichnis enthält den Eintrag `..` (Punkt Punkt) für das übergeordnete Verzeichnis, in dem es erstellt wurde, und den Eintrag `.` (Punkt) für das Verzeichnis selbst. In den meisten Verzeichnislisten sind diese Dateien verdeckt.

Verzeichnisstruktur

Die Struktur von Dateisystemen kann leicht sehr komplex werden. Es ist daher darauf zu achten, die Datei- und Verzeichnisstruktur so einfach wie möglich zu halten. Bei der Erstellung von Dateien und Verzeichnissen sollten leicht wiedererkennbare Namen vergeben werden. Dies erleichtert das Arbeiten mit Dateien.

Übergeordnetes Verzeichnis

Jedes Verzeichnis mit Ausnahme des Stammverzeichnisses (`/`) hat ein übergeordnetes Verzeichnis und kann untergeordnete Verzeichnisse enthalten.

Ausgangsverzeichnis

Nachdem Sie sich angemeldet haben, befinden Sie sich automatisch in Ihrem *Ausgangsverzeichnis* oder Anmeldeverzeichnis. Ein solches Verzeichnis wird vom Systemadministrator für jeden Benutzer eingerichtet. Ihr Ausgangsverzeichnis ist das Repository für Ihre persönlichen Dateien. Verzeichnisse, die Sie für die eigene Verwendung anlegen, sind in der Regel Unterverzeichnisse Ihres Ausgangsverzeichnisses. Wenn Sie in Ihr Ausgangsverzeichnis zurückkehren möchten, geben Sie an der Eingabeaufforderung den Befehl `cd` ein, und drücken Sie anschließend die Eingabetaste.

Arbeitsverzeichnis

Sie arbeiten immer in einem Verzeichnis. Das jeweilige Verzeichnis, in dem Sie arbeiten, wird als *aktuelles* Verzeichnis oder *Arbeitsverzeichnis* bezeichnet. Der Befehl `pwd` (Present Working Directory, aktuelles Arbeitsverzeichnis) gibt den Namen des Arbeitsverzeichnisses aus. Mit dem Befehl `cd` können Sie das Arbeitsverzeichnis wechseln.

Namenskonventionen für Verzeichnisse:

Die Namen der einzelnen Verzeichnisse müssen innerhalb des Verzeichnisses, in dem sie gespeichert sind, eindeutig sein. Damit ist gewährleistet, dass dem Verzeichnis auch ein eindeutiger Pfadname im Dateisystem zugeordnet ist.

Für Verzeichnisse gelten dieselben Namenskonventionen wie für Dateien. Lesen Sie hierzu den Abschnitt „Namenskonventionen für Dateien“ auf Seite 199.

Verzeichnisabkürzungen:

Abkürzungen sind gängige Spezifikationen für bestimmte Verzeichnisse.

Die folgende Liste enthält gültige Abkürzungen für Verzeichnisse:

| Abkürzung | Bedeutung |
|-----------|--|
| . | Das aktuelle Arbeitsverzeichnis. |
| .. | Das Verzeichnis oberhalb des aktuellen Arbeitsverzeichnisses (Elternverzeichnis des aktuellen Verzeichnisses). |
| ~ | Ihr Ausgangsverzeichnis. (Gilt nicht für die Bourne-Shell. Nähere Informationen finden Sie im Abschnitt „Bourne-Shell“ auf Seite 271.) |
| \$HOME | Ihr Ausgangsverzeichnis. (Gilt für alle Shells.) |

Pfadnamen von Verzeichnissen:

Jede Datei und jedes Verzeichnis kann über den eindeutigen Pfad, der als *Pfadname* bezeichnet wird, in der Baumstruktur des Dateisystems gefunden werden. Der Pfadname gibt die Position eines Verzeichnisses oder einer Datei in einem Dateisystem an.

Anmerkung: Pfadnamen dürfen nicht länger als 1023 Zeichen sein.

Im Dateisystem werden die folgenden Arten von Pfadnamen verwendet:

| Eintrag | Beschreibung |
|---------------------------|---|
| Absoluter Pfadname | Verfolgt den Pfad ausgehend vom Stammverzeichnis (/). Absolute Pfadnamen beginnen immer mit einem Schrägstrich (/). |
| Relativer Pfadname | Verfolgt den Pfad ausgehend vom aktuellen Verzeichnis über das Elternverzeichnis bzw. die Unterverzeichnisse und Dateien. |

Ein absoluter Pfadname ist der vollständige Name eines Verzeichnisses oder einer Datei ausgehend vom Stammverzeichnis (/). Unabhängig von Ihrem aktuellen "Standort" im Dateisystem können Sie mit dem absoluten Pfadnamen jedes Verzeichnis und jede Datei finden. Absolute Pfadnamen beginnen immer mit einem Schrägstrich (/) für das Stammverzeichnis. Der Pfadname /A/D/9 ist der absolute Pfadname für 9. Der erste Schrägstrich (/) steht für das Stammverzeichnis (/), d. h. das Ausgangsverzeichnis für den Suchvorgang. Der verbleibende Teil des Pfadnamens leitet die Suchoperation zu A, dann zu D und schließlich zu 9.

Es können durchaus zwei Dateien mit dem Namen 9 vorhanden sein, weil die absoluten Pfadnamen die Dateien im Dateisystem eindeutig machen. Die Pfadnamen /A/D/9 und /C/E/G/9 stehen für zwei eindeutige Dateien mit dem Namen 9.

Im Gegensatz zu den vollständigen Pfadnamen geben relative Pfadnamen ein Verzeichnis oder eine Datei ausgehend vom aktuellen Arbeitsverzeichnis an. Für relative Pfadnamen können Sie die Notation .. verwenden, um in der Dateisystemhierarchie aufwärts zu navigieren. Die Angabe .. steht für das Elternverzeichnis. Da relative Pfadnamen einen Pfad ausgehend vom aktuellen Verzeichnis bezeichnen, beginnen sie nicht mit einem Schrägstrich (/). Mit relativen Pfadnamen können Dateinamen im aktuellen Verzeichnis oder Pfadnamen von Dateien oder Verzeichnissen, die sich auf einer Ebene unterhalb oder oberhalb des aktuellen Verzeichnisses im Dateisystem befinden, angegeben werden. Wenn D das aktuelle Verzeichnis ist, muss als relativer Pfadname für den Zugriff auf 10 die Angabe F/10 verwendet werden. Der absolute Pfadname ist immer /A/D/F/10. Der relative Pfadname für den Zugriff auf 3 ist ../../B/3.

Sie können den Namen des aktuellen Verzeichnisses auch mit einem Punkt (.) angeben. Die Schreibweise mit Punkt (.) wird gewöhnlich bei der Ausführung von Programmen verwendet, die den aktuellen Verzeichnisnamen lesen.

Verzeichnisse erstellen (Befehl **mkdir**):

Mit dem Befehl **mkdir** können Sie Verzeichnisse erstellen, die Sie für den Parameter *Verzeichnis* angeben.

Jedes neue Verzeichnis enthält die Standardeinträge . (Punkt) und .. (Punkt Punkt). Die Berechtigungen für die neuen Verzeichnisse werden mit dem Flag **-m** *Modus* angegeben.

Wenn Sie ein Verzeichnis erstellen, wird es im aktuellen Verzeichnis (oder Arbeitsverzeichnis) erstellt, sofern Sie keinen absoluten Pfadnamen für eine andere Position im Dateisystem angeben.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **mkdir**:

- Geben Sie Folgendes ein, um im aktuellen Arbeitsverzeichnis ein neues Verzeichnis mit dem Namen Test und den Standardberechtigungen zu erstellen:

```
mkdir Test
```

- Geben Sie Folgendes ein, um ein Verzeichnis mit dem Namen Test mit den Berechtigungen `rw-r-xr-x` in einem zuvor erstellten Verzeichnis namens `/home/demo/sub1` zu erstellen:

```
mkdir -m 755 /home/demo/sub1/Test
```

- Geben Sie Folgendes ein, um ein Verzeichnis mit dem Namen Test mit den Standardberechtigungen im Verzeichnis `/home/demo/sub2` zu erstellen:

```
mkdir -p /home/demo/sub2/Test
```

Das Flag **-p** erstellt die Verzeichnisse `/home`, `/home/demo` und `/home/demo/sub2`, falls diese noch nicht vorhanden sind.

Die vollständige Syntax des Befehls **mkdir** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

Verzeichnisse verschieben oder umbenennen (Befehl mvdir):

Mit dem Befehl **mvdir** können Sie ein Verzeichnis verschieben oder umbenennen.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **mvdir**:

- Geben Sie Folgendes ein, um ein Verzeichnis zu verschieben:

```
mvdir book manual
```

Mit diesem Befehl wird das Verzeichnis `book` in das Verzeichnis `manual` verschoben, sofern das Verzeichnis `manual` vorhanden ist. Falls das Zielverzeichnis nicht vorhanden ist, wird das Verzeichnis `book` in `manual` umbenannt.

- Geben Sie Folgendes ein, um ein Verzeichnis zu verschieben und umzubenennen:

```
mvdir book3 proj4/manual
```

Wenn bereits ein Verzeichnis mit dem Namen `manual` vorhanden ist, wird `book3` samt Inhalt nach `proj4/manual` verschoben. Anders ausgedrückt, `book3` wird zu einem Unterverzeichnis von `proj4/manual`. Wenn `manual` nicht vorhanden ist, wird das Verzeichnis `book3` in `proj4/manual` umbenannt.

Die vollständige Syntax des Befehls **mvdir** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

Aktuelles Verzeichnis anzeigen (Befehl pwd):

Mit dem Befehl **pwd** können Sie den vollständigen Pfadnamen Ihres aktuellen Verzeichnisses (ausgehend vom Stammverzeichnis `/`) in die Standardausgabe schreiben.

Die einzelnen Verzeichnisse werden durch einen Schrägstrich (`/`) voneinander getrennt. Das Stammverzeichnis (`/`) wird durch den ersten Schrägstrich (`/`) dargestellt. Das zuletzt genannte Verzeichnis ist das aktuelle Verzeichnis.

Geben Sie Folgendes ein, um das aktuelle Verzeichnis anzuzeigen:

```
pwd
```

Der vollständige Pfadname des aktuellen Verzeichnisses wird wie folgt angezeigt:

/home/thomas

In ein anderes Verzeichnis wechseln (Befehl `cd`):

Mit dem Befehl `cd` können Sie vom aktuellen Verzeichnis in ein anderes Verzeichnis wechseln. Sie müssen Ausführungsberechtigung (Suchberechtigung) für das angegebene Verzeichnis besitzen.

Wenn Sie keinen Parameter *Verzeichnis* angeben, wechselt der Befehl `cd` in Ihr Anmeldeverzeichnis (`$HOME` in den Umgebungen `ksh` und `bsh` bzw. `$home` in der Umgebung `csch`). Das Verzeichnis wird zum aktuellen Verzeichnis, wenn Sie den vollständigen Pfadnamen des Verzeichnisses angeben. Ein vollständiger Pfadname beginnt mit einem Schrägstrich (`/`) für das Stammverzeichnis (`/`), einem Punkt (`.`) für das aktuelle Verzeichnis bzw. Punkt Punkt (`..`) für das übergeordnete Verzeichnis. Wenn der Verzeichnisname kein vollständiger Pfadname ist, sucht der Befehl `cd` das Verzeichnis relativ zu einem der Pfade, die mit der Shellvariablen `$CDPATH` (bzw. der `csch`-Variablen `$cdpath`) angegeben sind. Diese Variable hat dieselbe Syntax und eine ähnliche Semantik wie die Shellvariable `$PATH` (bzw. die `csch`-Variable `$path`).

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls `cd`:

- Geben Sie Folgendes ein, um in das Ausgangsverzeichnis zu wechseln:

```
cd
```

- Geben Sie Folgendes ein, um in das Verzeichnis `/usr/include` zu wechseln:

```
cd /usr/include
```

- Geben Sie Folgendes ein, um in der Verzeichnisstruktur eine Ebene tiefer, in das Verzeichnis `sys`, zu wechseln:

```
cd sys
```

Wenn das aktuelle Verzeichnis `/usr/include` ist und ein Unterverzeichnis mit dem Namen `sys` enthält, wird `/usr/include/sys` zum aktuellen Verzeichnis.

- Geben Sie Folgendes ein, um in der Verzeichnisstruktur eine Ebene höher zu wechseln:

```
cd ..
```

Der spezielle Dateiname `..` (Punkt Punkt) bezeichnet das Verzeichnis, das dem aktuellen Verzeichnis direkt übergeordnet ist.

Die vollständige Syntax des Befehls `cd` ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

Verzeichnisse kopieren (Befehl `cp`):

Mit dem Befehl `cp` können Sie den Inhalt der bzw. des mit dem Parameter *Quelldatei* bzw. *Quellenverzeichnis* angegebenen Datei bzw. Verzeichnisses in die bzw. das mit dem Parameter *Zieldatei* bzw. *Zielverzeichnis* angegebene Datei bzw. Verzeichnis kopieren.

Falls die angegebene *Zieldatei* vorhanden ist, wird der Originalinhalt der Zieldatei ohne Warnung überschrieben. Wenn Sie mehrere *Quelldateien* kopieren, muss das Ziel ein Verzeichnis sein.

Wenn Sie die *Quelldatei* in ein Verzeichnis kopieren möchten, müssen Sie den Pfad eines vorhandenen *Zielverzeichnisses* angeben. Dateien behalten ihren Namen, wenn sie in ein anderes Verzeichnis kopiert werden, sofern am Ende des Pfads kein neuer Dateiname angegeben wird. Der Befehl `cp` kopiert auch vollständige Verzeichnisse in andere Verzeichnisse, wenn Sie das Flag `-r` oder `-R` angeben.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls `cp`:

- Geben Sie Folgendes ein, um alle Dateien aus dem Verzeichnis `/home/accounts/customers/orders` in das Verzeichnis `/home/accounts/customers/shippments` zu kopieren:

```
cp /home/accounts/customers/orders/* /home/accounts/customers/shippments
```

Dieser Befehl kopiert die Dateien, aber nicht die Verzeichnisse aus dem Verzeichnis `orders` in das Verzeichnis `shipments`.

- Geben Sie Folgendes ein, um ein Verzeichnis einschließlich aller Dateien und Unterverzeichnisse in ein anderes Verzeichnis zu kopieren:

```
cp -R /home/accounts/customers /home/accounts/vendors
```

Dieser Befehl kopiert das Verzeichnis `customers` einschließlich aller Dateien, Unterverzeichnisse und der Dateien in diesen Unterverzeichnissen in das Verzeichnis `vendors`.

Die vollständige Syntax des Befehls `cp` ist in der Veröffentlichung *Commands Reference, Volume 1* beschrieben.

Inhalt eines Verzeichnisses anzeigen (Befehl `ls`):

Mit dem Befehl `ls` können Sie den Inhalt eines Verzeichnisses anzeigen.

Der Befehl `ls` schreibt den Inhalt jedes angegebenen *Verzeichnisses* oder jeder angegebenen *Datei* zusammen mit allen anderen Informationen, die mit den Flags angefordert werden, in die Standardausgabe. Wenn Sie keine *Datei* und kein *Verzeichnis* angeben, zeigt der Befehl `ls` den Inhalt des aktuellen Verzeichnisses an.

Standardmäßig zeigt der Befehl `ls` alle Informationen in alphabetischer Reihenfolge sortiert nach Dateinamen an. Wenn der Befehl von einem Benutzer mit Rootberechtigung ausgeführt wird, verwendet er standardmäßig das Flag `-A`, das alle Einträge außer Punkt (`.`) und Punkt Punkt (`..`) auflistet. Wenn Sie alle Einträge für Dateien, einschließlich derer, die mit einem Punkt (`.`) beginnen, anzeigen möchten, verwenden Sie den Befehl `ls -a`.

Sie können die Ausgabe auf die folgenden Arten formatieren:

- Pro Zeile einen Eintrag auflisten. Verwenden Sie hierfür das Flag `-l`.
- Einträge in mehreren Spalten auflisten. Verwenden Sie hierfür das Flag `-C` oder das Flag `-x`. Das Flag `-C` ist das Standardformat, wenn die Ausgabe auf einem nicht grafikfähigen Terminal (TTY) erfolgt.
- Einträge durch Kommas voneinander getrennt auflisten. Verwenden Sie hierfür das Flag `-m`.

Der Befehl `ls` verwendet die Umgebungsvariable `$COLUMNS`, um die Anzahl der Zeichenpositionen in der Ausgabezeile zu bestimmen. Wenn diese Variable nicht gesetzt ist, liest der Befehl die Datei `terminfo`. Falls der Befehl `ls` die Anzahl der Zeichenpositionen mit keiner dieser Methoden bestimmen kann, verwendet er den Standardwert 80.

Die mit den Flags `-e` und `-l` angezeigten Informationen werden wie folgt interpretiert:

Das erste Zeichen kann Folgendes sein:

| Eintrag | Beschreibung |
|----------------|--------------------------------------|
| <code>d</code> | Ein Verzeichnis. |
| <code>b</code> | Eine blockorientierte Gerätedatei. |
| <code>c</code> | Eine zeichenorientierte Gerätedatei. |
| <code>l</code> | Eine symbolische Verbindung. |
| <code>p</code> | Eine FIFO-Gerätedatei. |
| <code>s</code> | Ein lokaler Socket. |
| <code>-</code> | Eine gewöhnliche Datei. |

Die folgenden neun Zeichen werden in drei Gruppen zu je drei Zeichen eingeteilt. Die ersten drei Zeichen geben die Berechtigung des Datei- bzw. Verzeichniseigners an. Die nächsten drei Zeichen geben die Berechtigung der anderen Benutzer in der Gruppe an. Die letzten drei Zeichen geben die Berechtigung al-

ler anderen Benutzer an, die auf die Datei zugreifen können. Die drei Zeichen in den einzelnen Gruppen geben die Lese-, Schreib- und Ausführungsberechtigung für die Datei an. Mit der Ausführungsberechtigung für ein Verzeichnis kann innerhalb des Verzeichnisses nach einer bestimmten Datei gesucht werden.

Die Berechtigungen werden wie folgt angezeigt:

| Eintrag | Beschreibung |
|---------|--|
| r | Leseberechtigung erteilt |
| t | Nur der Verzeichniseigner bzw. der Dateieigner kann eine Datei innerhalb dieses Verzeichnisses löschen oder umbenennen, selbst wenn andere Benutzer über Schreibberechtigung für das Verzeichnis verfügen. |
| w | Schreib-/Editierberechtigung erteilt |
| x | Ausführungs-/Suchberechtigung erteilt |
| - | Entsprechende Berechtigung nicht erteilt |

Das Flag **-e** zeigt bis auf ein zusätzliches elftes Zeichen dieselben Informationen an wie das Flag **-l**. Dieses elfte Zeichen wird wie folgt interpretiert:

| Eintrag | Beschreibung |
|---------|--|
| + | Gibt an, dass eine Datei erweiterte Sicherheitsinformationen enthält. Die Datei verfügt in diesem Modus beispielsweise über das erweiterte Attribut ACL , TCB oder TP . |
| - | Gibt an, dass die Datei keine erweiterten Sicherheitsinformationen enthält. |

Wenn die Größe der Dateien in einem Verzeichnis aufgelistet wird, zeigt der Befehl **ls** die Gesamtanzahl der Blöcke einschließlich der indirekten Blöcke an.

Beispiele:

- Geben Sie Folgendes ein, um alle Dateien im aktuellen Verzeichnis aufzulisten:

```
ls -a
```

Es werden alle Dateien aufgelistet, einschließlich

- Punkt (.)
 - Punkt Punkt (..)
 - und anderen Dateien, deren Name mit einem Punkt (.) beginnt oder nicht.
- Geben Sie Folgendes ein, um detaillierte Informationen anzuzeigen:

```
ls -l chap1 .profile
```

Es wird eine ausführliche Auflistung mit detaillierten Informationen zu `chap1` und `.profile` angezeigt.

- Geben Sie Folgendes ein, um detaillierte Informationen zu einem Verzeichnis anzuzeigen:

```
ls -d -l . manual manual/chap1
```

Dieser Befehl zeigt eine ausführliche Liste für die Verzeichnisse `.` und `manual` sowie für die Datei `manual/chap1` an. Wenn Sie das Flag **-d** nicht angeben, werden nur die Dateien in den Verzeichnissen `.` und `manual` anstelle der detaillierten Informationen zu den Verzeichnissen selbst angezeigt.

Die vollständige Syntax des Befehls **ls** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

Verzeichnisse löschen oder entfernen (Befehl **rmdir**):

Mit dem Befehl **rmdir** können Sie das mit dem Parameter *Verzeichnis* angegebene Verzeichnis vom System entfernen.

Das Verzeichnis muss leer sein (es darf nur `.` und `..` enthalten), damit Sie es entfernen können. Außerdem müssen Sie Schreibberechtigung für das Elternverzeichnis besitzen. Verwenden Sie den Befehl `ls -a Verzeichnis`, um zu überprüfen, ob das Verzeichnis leer ist.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **rmdir**:

- Geben Sie Folgendes ein, um ein Verzeichnis zu leeren und zu entfernen:

```
rm meinverz/* meinverz/.  
rmdir meinverz
```

Erst wird der Inhalt des Verzeichnisses `meinverz` gelöscht und anschließend das leere Verzeichnis entfernt. Der Befehl `rm` weist Sie in einer Fehlermeldung darauf hin, dass er versucht, die Verzeichnisse `.` und `..` zu entfernen. Anschließend entfernt der Befehl **rmdir** diese Verzeichnisse und das leere Verzeichnis selbst.

Anmerkung: Der Befehl `rm meinverz/* meinverz/.*` entfernt zuerst die Dateien mit Namen, die nicht mit einem Punkt beginnen, und dann die Dateien, die mit einem Punkt beginnen. Der Befehl `ls` listet Dateinamen, die mit einem Punkt beginnen nur dann auf, wenn Sie das Flag `-a` verwenden.

- Geben Sie Folgendes ein, um das Verzeichnis `/tmp/jones/demo/meinverz` und alle darin enthaltenen Unterverzeichnisse zu entfernen:

```
cd /tmp  
rmdir -p jones/demo/meinverz
```

Das Verzeichnis `jones/demo/meinverz` wird aus dem Verzeichnis `/tmp` entfernt. Wenn ein Verzeichnis nicht leer ist oder Sie keine Schreibberechtigung für das zu entfernende Verzeichnis besitzen, wird der Befehl mit entsprechenden Fehlermeldungen beendet.

Die vollständige Syntax des Befehls **rmdir** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

Inhalt von Verzeichnissen vergleichen (Befehl **dircmp**):

Mit dem Befehl **dircmp** können Sie zwei Verzeichnisse, die Sie mit den Parametern *Verzeichnis1* und *Verzeichnis2* angeben, vergleichen und die Ergebnisse des Vergleichs in die Standardausgabe schreiben.

Der Befehl **dircmp** vergleicht zuerst die Dateinamen in den Verzeichnissen. Falls eine Datei in beiden Verzeichnissen enthalten ist, vergleicht der Befehl **dircmp** den Inhalt der beiden Dateien.

Der Befehl **dircmp** listet in seiner Ausgabe die Dateien auf, die jeweils in nur einem der Verzeichnisse vorhanden sind. Anschließend werden die Dateien aufgelistet, die in beiden Verzeichnissen unter demselben Namen, aber mit unterschiedlichem Inhalt vorhanden sind. Wird kein Flag angegeben, werden zusätzlich die Dateien mit identischem Namen und Inhalt in beiden Verzeichnissen aufgelistet.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **dircmp**:

- Geben Sie beispielsweise Folgendes ein, um die Unterschiede zwischen den Dateien `proj.ver1` und `proj.ver2` zusammenzufassen:

```
dircmp proj.ver1 proj.ver2
```

Es erscheint eine Zusammenfassung der Unterschiede zwischen den Verzeichnissen `proj.ver1` und `proj.ver2`. In der Zusammenfassung werden die Dateien, die nur in einem der beiden Verzeichnisse vorhanden sind, und die Dateien, die in beiden vorhanden sind, getrennt aufgelistet. Wird eine Datei in beiden Verzeichnissen gefunden, stellt der Befehl **dircmp** fest, ob die beiden Kopien identisch sind.

- Geben Sie beispielsweise Folgendes ein, um Detailinformationen zu den Unterschieden in den Verzeichnissen `proj.ver1` und `proj.ver2` anzuzeigen:

```
dircmp -d -s proj.ver1 proj.ver2
```

Das Flag **-s** unterdrückt Informationen zu identischen Dateien. Das Flag **-d** zeigt eine **diff**-Auflistung aller unterschiedlichen Dateien aus beiden Verzeichnissen an.

Die vollständige Syntax des Befehls **dircmp** ist in der Veröffentlichung *Commands Reference, Volume 2* beschrieben.

Befehlsübersicht für Dateisysteme und Verzeichnisse:

Im Folgenden finden Sie eine Übersicht über die Befehle für Dateisysteme und Verzeichnisse, über die Befehle für die Bearbeitung von Verzeichnissen und eine Liste mit Verzeichnisabkürzungen.

Tabelle 66. Befehlsübersicht für Dateisysteme

| Eintrag | Beschreibung |
|-----------|--|
| df | Zeigt Informationen zum Speicherplatz in Dateisystemen an. |

Tabelle 67. Verzeichnisabkürzungen

| Eintrag | Beschreibung |
|---------------|--|
| . | Das aktuelle Arbeitsverzeichnis. |
| .. | Das Elternverzeichnis des aktuellen Arbeitsverzeichnisses. |
| ~ | Ihr Ausgangsverzeichnis. (Gilt nicht für die Bourne-Shell. Nähere Informationen finden Sie im Abschnitt „Bourne-Shell“ auf Seite 271.) |
| \$HOME | Ihr Ausgangsverzeichnis. (Gilt für alle Shells.) |

Tabelle 68. Befehlsübersicht für die Bearbeitung von Verzeichnissen

| Eintrag | Beschreibung |
|---------------|---|
| cd | Wechselt das aktuelle Verzeichnis. |
| cp | Kopiert Dateien oder Verzeichnisse. |
| dircmp | Vergleicht zwei Verzeichnisse und den Inhalt ihrer gemeinsamen Dateien. |
| ls | Zeigt den Inhalt eines Verzeichnisses an. |
| mkdir | Erstellt Verzeichnisse. |
| mvdir | Verschiebt Verzeichnisse (benennt sie um). |
| pwd | Zeigt den Pfadnamen des Arbeitsverzeichnisses an. |
| rmdir | Entfernt ein Verzeichnis. |

Workload Manager

Mit Workload Manager (WLM) kann der Systemadministrator wirkungsvoller steuern, wie der VMM (Virtual Memory Manager) des Schedulers und das Platten-E/A-Subsystem Prozessen Ressourcen zuordnen.

Sie können WLM verwenden, um zu verhindern, dass unterschiedliche Klassen von Jobs miteinander in Konflikt geraten, und Sie können Ressourcen basierend auf den Anforderungen unterschiedlicher Benutzergruppen zuordnen.

Achtung: Die effiziente Nutzung von WLM setzt umfassende Kenntnisse der vorhandenen Systemprozesse und der Systemleistung voraus. Wenn der Systemadministrator WLM mit extremen oder ungenauen Werten konfiguriert, hat dies eine erhebliche Verschlechterung der Leistung zur Folge.

WLM ist primär für den Einsatz in Großsystemen bestimmt. Großsysteme werden häufig für die Serverkonsolidierung verwendet, bei der die Workloads vieler unterschiedlicher Serversysteme (z. B. Drucker, Datenbank, allgemeine Benutzersysteme und Transaktionsverarbeitungssysteme) zu einem einzigen Großsystem kombiniert werden, um die Kosten der Systemverwaltung zu senken. Diese Workloads geraten häufig miteinander in Konflikt und haben unterschiedliche Ziele und Servicevereinbarungen.

WLM unterstützt außerdem die Isolierung von Benutzergruppen mit stark abweichendem Systemverhalten. Auf diese Weise kann nachhaltig die Blockierung von Workloads mit bestimmten Verhalten (z. B. in-

teraktive Jobs oder Jobs mit geringer CPU-Lastung) durch Workloads mit anderen Verhalten (z. B. Stapeljobs oder Jobs mit hoher Speicherlastung) verhindert werden.

Außerdem ist WLM an das Abrechnungssystem angebunden und ermöglicht Benutzern damit, zusätzlich zur Standardabrechnung pro Benutzer oder Gruppe Abrechnungen zur Ressourcennutzung pro WLM-Klasse durchzuführen.

Zugehörige Konzepte:

„Systemabrechnung“ auf Seite 164

Mit dem Dienstprogramm für Systemabrechnung können Sie Informationen zur Verwendung verschiedener Systemressourcen für Einzelpersonen und Gruppen erfassen und zurückmelden.

Workload-Management-Konzepte

Mit WLM können Sie unterschiedliche Serviceklassen für Jobs erstellen und Attribute für diese Klassen festlegen.

Diese Attribute legen die Mindest- und Maximalwerte für CPU, physischen Speicher und Platten-E/A-Durchsatz für eine Klasse fest. WLM ordnet die Jobs anschließend unter Verwendung der Klassenzuordnungsregeln, die von einem Systemadministrator bereitgestellt werden, automatisch Klassen zu. Diese Zuordnungsregeln basieren auf den Werten einer Gruppe von Attributen für einen Prozess. Der Systemadministrator oder ein berechtigter Benutzer kann Jobs auch manuell zu Klassen zuordnen und damit die automatische Zuordnung überschreiben.

Zugehörige Konzepte:

„Workload Manager verwalten“ auf Seite 522

Mit Workload Manager (WLM) haben Systemadministratoren eine erheblich bessere Kontrolle darüber, wie der Scheduler und der Virtual Memory Manager (VMM) Ressourcen für Prozesse reservieren. Sie können mit WLM verhindern, dass unterschiedliche Klassen von Jobs miteinander in Konflikt geraten, und Sie können Ressourcen basierend auf den Anforderungen unterschiedlicher Benutzergruppen zuordnen.

Terminologie für das Workload-Management:

In dieser Tabelle sind allgemeine Begriffe, die im Kontext des Workload-Management verwendet werden, aufgelistet und beschrieben.

| Eintrag | Beschreibung |
|----------------|---|
| Klasse | Eine <i>Klasse</i> ist eine Sammlung von Prozessen und den ihnen zugehörigen Threads. Eine Klasse hat eine eigene Gruppe von Ressourcengrenzwerten und Zielanteilen (Share). Der Begriff <i>Klasse</i> wird verwendet, um Unterklassen und Superklassen zu beschreiben. |
| Superklasse | Eine <i>Superklasse</i> ist eine Klasse, der Unterklassen zugeordnet sind. Prozesse können nicht zu einer Superklasse gehören, ohne gleichzeitig auch zu einer Unterklasse zu gehören. Eine Superklasse hat eine Gruppe von Zuordnungsregeln, die bestimmen, welche Prozesse der Superklasse zugeordnet werden. Eine Superklasse hat außerdem eine Gruppe von Ressourcengrenzwerten und Ressourcenzielanteilen, die bestimmen, welche Ressourcenmenge von den Prozessen, die zu der Superklasse gehören, verbraucht werden kann. Diese Ressourcen werden basierend auf den Ressourcengrenzwerten und Ressourcenzielanteilen der Unterklassen auf die Unterklassen verteilt. |

| Eintrag | Beschreibung |
|--|--|
| Unterklassen | <p>Eine <i>Unterklasse</i> ist eine Klasse, die genau einer Superklasse zugeordnet ist. Jeder Prozess in einer Unterklasse ist auch Mitglied der zugehörigen Superklasse. Unterklassen können nur auf die Ressourcen zugreifen, die der Superklasse zur Verfügung stehen. Eine Unterklasse hat eine Gruppe von Klassenzuordnungsregeln, die bestimmen, welche Prozesse, die der Superklasse zugeordnet sind, zur Unterklasse gehören. Eine Unterklasse hat außerdem eine Gruppe von Ressourcengrenzwerten und Ressourcenzielanteilen, die bestimmen, welche Ressourcenmenge von den Prozessen in der Unterklasse verbraucht werden können.</p> <p>Diese Ressourcengrenzwerte und Ressourcenzielanteile geben an, welche Menge der Ressourcen, die der Superklasse zur Verfügung stehen (Ziel für die Superklasse) von Prozessen in der Unterklasse verbraucht werden kann.</p> <p>Für die WLM-Verwaltung kann SMIT oder die WLM-Befehlszeilenschnittstelle verwendet werden.</p> |
| Klassifizierungsmechanismus | Ein <i>Klassifizierungsmechanismus</i> ist eine Gruppe von Klassenzuordnungsregeln, die bestimmen, welche Prozesse welchen Klassen zugeordnet werden (Superklassen oder Unterklassen innerhalb von Superklassen). |
| Klassenzuordnungsregel | Eine <i>Klassenzuordnungsregel</i> gibt an, welche Werte in einer Gruppe von Prozessattributen dazu führen, dass ein Prozess einer bestimmten Klasse (Superklasse oder Unterklasse in einer Superklasse) zugeordnet werden. |
| Prozessattributwert | Ein <i>Prozessattributwert</i> ist der Wert, den ein Prozess für ein Prozessattribut verwendet. Zu den Prozessattributen gehören Attribute wie die Benutzer-ID, die Gruppen-ID und der Name des Anwendungspfads. |
| Ressourcengrenzwerte | <i>Ressourcengrenzwerte</i> sind eine Gruppe von Werten, die WLM für eine Gruppe von Ressourcenauslastungswerten verwaltet. Diese Grenzwerte sind vollständig unabhängig von den Ressourcengrenzen, die mit der Subroutine setrlimit angegeben werden. |
| Ressourcenzielanteil | <i>Ressourcenzielanteile</i> sind die Anteile einer Ressource, die einer Klasse (Superklasse oder Unterklasse) zur Verfügung stehen. Diese Anteile werden zusammen mit anderen Klassenanteilen (Superklasse oder Unterklasse) derselben Ebene und Schicht verwendet, um die gewünschte Verteilung der Ressourcen auf die Klassen dieser Ebene und Schicht zu bestimmen. |
| Ressourcenauslastungswert | Ein <i>Ressourcenauslastungswert</i> gibt die Menge einer Ressource an, die ein Prozess oder eine Prozessgruppe derzeit in einem System verwendet. Ob es sich um einen Prozess oder eine Prozessgruppe handelt, richtet sich nach dem Geltungsbereich der Prozessressourcensammlung. |
| Geltungsbereich der Ressourcensammlung | Der <i>Geltungsbereich der Ressourcensammlung</i> ist die Ebene, auf der die Ressourcenauslastung erfasst wird, und die Ebene, auf der Ressourcengrenzwerte angewendet werden. Die folgenden Ebenen sind möglich: einzelner Prozess in einer Klasse, Summe aller Prozesse in einer Klasse pro Benutzer oder Summe aller Prozesse in einer Klasse. Derzeit wird nur der zuletzt genannte Geltungsbereich unterstützt. |
| Prozessklasseneigenschaften | Die <i>Prozessklasseneigenschaften</i> sind die Gruppe von Eigenschaften, die ein Prozess basierend auf den Klassen (Unterklasse und Superklasse), denen er zugeordnet ist, erhält. |
| Klassenberechtigungen | Die <i>Klassenberechtigungen</i> sind eine Gruppe von Regeln, die angeben, welche Benutzer und Gruppen Operationen für eine Klasse oder für Prozesse und Threads in einer Klasse ausführen dürfen. Hierzu gehört unter anderem die Berechtigung, Prozesse manuell einer Klasse zuzuordnen oder Unterklassen einer Superklasse zu erstellen. |
| Klassenschicht | Die <i>Klassenschicht</i> (Tier) ist die Position der Klasse in der Hierarchie der gewünschten Ressourcenbeschränkung für alle Klassen. Die Ressourcengrenzen (einschließlich der Ressourcenziele) für alle Klassen in einer Schicht werden ausgeschöpft, bevor eine Ressource Klassen niedrigerer Schichten zur Verfügung gestellt wird. Schichten werden auf Superklassen- und Unterklassenebene unterstützt. Ressourcen werden Superklassen basierend auf ihren Schichten bereitgestellt. In einer Superklasse werden Ressourcen Unterklassen basierend auf den Schichtwerten der Unterklassen innerhalb der Superklasse zugeordnet. Somit ist die Superklassenschicht das wichtigste Differenzierungsmerkmal für die Ressourcenverteilung. Die Unterklassenschicht ist ein zusätzliches kleineres Differenzierungsmerkmal innerhalb einer Superklasse. |

Klassen für das Workload-Management:

WLM ermöglicht Systemadministratoren, Klassen und für jede Klasse eine Gruppe von Attributen und Ressourcengrenzen zu definieren.

Die Prozesse werden basierend auf den vom Systemadministrator definierten Kriterien Klassen zugeordnet. Die Nutzungsrechte und Grenzwerte für Ressourcen werden auf Klassenebene umgesetzt. Durch die Definition von Serviceklassen und die Regulierung der Ressourcenauslastung für jede Klasse von Anwendungen wird verhindert, dass sich Anwendungen mit sehr unterschiedlichen Ressourcennutzungsmustern gegenseitig behindern, wenn sie auf einem Server ausgeführt werden.

WLM unterstützt eine Hierarchie von Klassen mit zwei Ebenen:

- Die Ressourcen des Systems werden gemäß den Ressourcennutzungsrechten der Superklassen auf die Superklassen verteilt. Der Systemadministrator definiert die Nutzungsrechte für Ressourcen.
- Jede Superklasse kann Unterklassen haben. Die der Superklasse zugeordneten Ressourcen werden gemäß den Ressourcennutzungsrechten der einzelnen Unterklassen auf die Unterklassen verteilt.
- Der Systemadministrator kann die Verwaltung der Unterklassen jeder Superklasse an einen *Superklassenadministrator* oder an eine Gruppe von Superklassenadministratoren delegieren.
- WLM unterstützt bis zu 69 Superklassen (64 benutzerdefinierte) und 64 Unterklassen pro Superklasse (61 benutzerdefinierte).
- Je nach Anforderungen der Organisation kann ein Systemadministrator entscheiden, nur Superklassen oder Superklassen und Unterklassen zu verwenden.

Anmerkung: In dieser Beschreibung von WLM bezieht sich der Begriff *Klasse* durchgängig auf Superklassen und Unterklassen. Wenn sich eine Beschreibung ausschließlich auf einen bestimmten Klassentyp bezieht, ist der Typ explizit angegeben.

Zuordnung von Prozessen zu Klassen für das Workload-Management:

Die Prozesse werden unter Verwendung der vom Systemadministrator definierten Klassenzuordnungsregeln einer Klasse zugeordnet. Die Klassifizierungskriterien basieren auf dem Wert einer Gruppe von Prozessattributen, wie z. B. der Benutzer-ID, der Gruppen-ID, dem Namen der Anwendungsdatei, dem Prozessstyp oder der Prozesskennung.

Zur Bestimmung der Superklasse, der ein Prozess zugeordnet wird, wird eine definierte Gruppe von Regeln verwendet. Wenn für diese Superklasse Unterklassen definiert sind, gibt es eine weitere Gruppe von Regeln für diese Superklasse, mit der bestimmt werden kann, welche Unterklasse welchem Prozess zugeordnet wird. Dieser automatische Zuordnungsprozess berücksichtigt auch die Vererbungsattribute (**inheritance**) von Superklasse und Unterklasse.

Die automatische Klassenzuordnung wird durchgeführt, wenn ein Prozess die Subroutine **exec** aufruft. Die Klassenzuordnung wird erneut geprüft, wenn ein Prozess eine Subroutine verwendet, die ein Prozessattribut, das für die Klassifizierung verwendet wird, ändern kann. Beispiele hierfür sind die Subroutinen **setuid**, **setgid**, **setpri** und **plock**.

Zusätzlich zu dieser automatischen Klassenzuordnung kann ein Benutzer mit entsprechender Berechtigung Prozesse oder Prozessgruppen manuell einer bestimmten Superklasse oder Unterklasse zuordnen.

Zugehörige Konzepte:

„Klassenattribute“ auf Seite 537

Sie können alle Attribute einer WLM-Klasse auflisten.

Ressourcensteuerung:

Mit WLM können Ressourcen auf zwei Arten verwaltet werden: als Prozentsatz der verfügbaren Ressourcen oder als Gesamtressourcenauslastung.

Die folgenden Ressourcen können auf Prozentbasis gesteuert werden:

- Prozessorauslastung durch die Threads des Typs SCHED_OTHER in einer Klasse. Dies ist die Summe aller Prozessorzyklen, die von allen Threads in einer Klasse verbraucht wurden. Threads mit fester Priorität können nicht angepasst werden. Sie können deshalb nicht geändert werden und sie können das Ziel für die Prozessorauslastung überschreiten.
- Auslastung des physischen Hauptspeichers durch die Prozesse in einer Klasse. Dies ist die Summe aller Speicherseiten, die zu den Prozessen in der Klasse gehören.
- Platten-E/A-Bandbreite der Klasse. Dies ist die Bandbreite (in 512-Byte-Blöcken pro Sekunde) aller E/As, die von Threads in der Klasse auf jeder Platteneinheit, auf die die Klasse zugreift, gestartet werden.

Ressourcen, die auf Basis der Gesamtauslastung gesteuert werden können, gehören zu einer von zwei Kategorien: Klassengesamtsummen und Prozessgesamtsummen. Die Kategorie Klassengesamtsummen umfasst Folgendes:

Anzahl der Prozesse in einer Klasse

Dies ist die Anzahl der Prozesse, die gleichzeitig in einer Klasse aktiv sind.

Anzahl der Threads in einer Klasse

Dies ist die Anzahl der Threads, die gleichzeitig in einer Klasse aktiv sind.

Anzahl der Anmeldungen in einer Klasse

Dies ist die Anzahl der Anmeldungen, die gleichzeitig in einer Klasse aktiv sind.

Die Kategorie Prozessgesamtsummen umfasst Folgendes:

Gesamt-CPU-Zeit

Dies ist die von einem einzelnen Prozess verbrauchte Gesamt-CPU-Zeit.

Gesamtplatten-E/A

Dies ist die Gesamtanzahl aller E/As für Plattenblöcke eines einzelnen Prozesses.

Gesamtverbindungsdauer

Dies ist die Gesamtdauer, die eine Anmeldesitzung aktiv sein kann.

Nutzungsrechte für Ressourcen:

Mit WLM können Systemadministratoren für jeden einzelnen Ressourcentyp Nutzungsrechte für Ressourcen pro Klasse festlegen.

Diese Nutzungsrechte können mit den folgenden Angaben festgelegt werden:

- Ziel für die Nutzung unterschiedlicher Ressourcentypen. Dieses Ziel wird in Form von Anteilen definiert. Die Anteile werden als relative Nutzungszahlen zwischen unterschiedlichen Klassen angegeben. Wenn beispielsweise zwei Klassen 1 bzw. 3 CPU-Anteile haben und derzeit die einzigen beiden aktiven Klassen sind, ist das von WLM für die CPU-Regulierung verwendete Ziel für diese Klassen 25 % bzw. 75 %. Die prozentualen Zielwerte werden für die Klassen in jeder Schicht basierend auf der Anzahl aktiver Anteile in der Schicht und der verfügbaren Ressourcenmenge für die Schicht berechnet.
- Mindest- und Maximalwerte. Diese Grenzwerte werden als Prozentsatz der verfügbaren Gesamtressourcenmenge angegeben. WLM unterstützt zwei Arten von Maximalwerten:
 - Ein veränderlicher Maximalwert gibt die maximale Ressourcenmenge an, die zur Verfügung gestellt werden kann, wenn eine Konkurrenzsituation um die Ressource eintritt. Dieser Maximalwert kann überschritten werden, wenn keine Konkurrenzsituation vorliegt, d. h., wenn die Ressource von niemand anderem angefordert wird.
 - Ein fester Maximalwert gibt die maximale Ressourcenmenge an, die zur Verfügung gestellt werden kann, unabhängig davon, ob es eine Konkurrenzsituation um die Ressource gibt. Threads mit fester Priorität unterliegen jedoch nicht denselben Regeln und können deshalb den Grenzwert überschreiten.

- **Gesamtgrenzwerte.** Die Gesamtgrenzwerte werden streng eingehalten. Wenn ein Prozess einen seiner Gesamtverbrauchsgrenzwerte überschreitet, wird er beendet. Hat eine Klasse einen ihrer Gesamtgrenzwerte erreicht, scheitert jede Operation, die zum Erstellen einer weiteren Instanz dieser Ressource führen würde.

In den meisten Fällen reichen variable Maximalwerte aus, um sicherzustellen, dass die Nutzungsrechte für Ressourcen eingehalten und umgesetzt werden. Die Verwendung fester Maximalwerte kann nicht genutzte Systemressourcen zur Folge haben, da diese Maximalwerte strikt umgesetzt werden, selbst wenn es keine Konkurrenzsituation um die Ressource gibt. Es sind sorgfältige Überlegungen in Bezug auf die Verwendung fester Maximalwerte anzustellen, da zu niedrig gewählte Werte erhebliche Auswirkungen auf die System- oder Anwendungsleistung haben können. Auch Gesamtgrenzwerte sollten mit Vorsicht verwendet werden, da diese dazu führen können, dass Prozesse beendet werden oder nicht wie beabsichtigt arbeiten.

Im aktiven Modus versucht WLM, aktive Klassen nahe an ihren Zielwerten zu halten. Da die Werte für die verschiedenen Grenzwerte einigen Einschränkungen unterliegen, kann die Summe aller Grenzwerte für alle Klassen die 100-Prozent-Marke weit überschreiten. Wenn alle Klassen aktiv sind, kann der Grenzwert nicht von allen Klassen erreicht werden. WLM reguliert den Prozessorverbrauch, indem er für die Anpassung der Planungsprioritäten der Threads mit nicht festen Prioritäten im System die Leistung der Klasse, zu der die Threads gehören, in Relation zu ihren Grenz- und Zielwerten zugrundelegt. Dieser Ansatz gewährleistet einen durchschnittlichen Prozessorverbrauch über einen bestimmten Zeitraum und nicht nur für ein sehr kurzes Intervall (z. B. 10 ms).

Wenn Klasse A beispielsweise die einzige aktive Klasse mit einem Prozessmindestwert von 0 % und einem Prozessorzielwert von 60 Anteilen ist, erhält sie 100 % der Prozessorleistung. Wenn Klasse B mit einem Prozessormindestwert von 0 % und einem Prozessorzielwert von 40 Anteilen aktiviert wird, verringert sich die Prozessorauslastung durch Klasse A nach und nach auf 60 %, und die Prozessorauslastung durch Klasse B steigt von 0 % auf 40 %. Das System pendelt sich in nur wenigen Sekunden bei einer Prozessorauslastung von jeweils 60 % und 40 % ein.

In diesem Beispiel wird vorausgesetzt, dass keine Konkurrenzsituationen um den Hauptspeicher zwischen den Klassen vorliegen. Im regulären Betrieb sind die Grenzwerte, die Sie für Prozessor und Hauptspeicher festlegen, voneinander abhängig. Eine Klasse ist möglicherweise nicht in der Lage, ihren Zielwert oder nicht einmal ihre Mindestprozessorzuordnung zu erreichen, wenn ihr Maximalwert für die Speicherbelegung im Vergleich mit ihrem Ausführungsspeicher (Working Set) zu klein ist.

Für die Optimierung der Klassendefinition und Klassengrenzwerte für eine bestimmte Gruppe von Anwendungen stellt WLM das Berichtstool **wlmstat** bereit, das die derzeit von jeder Klasse verwendete Ressourcenmenge anzeigt. Für die Systemüberwachung wird außerdem das grafisch orientierte Anzeigetool **wlmon** bereitgestellt.

Virtuellen Speicher mit Workload Manager begrenzen:

Mit WLM-Grenzwerten (Workload Manager) für virtuellen Speicher können Administratoren eine Verschlechterung der Systemleistung oder einen Systemausfall aufgrund exzessiven Pagings verhindern, indem sie einen Grenzwert für den virtuellen Speicher für eine Klasse oder einen Prozess festlegen.

Wenn ein Grenzwert überschritten wird, führt WLM eine der folgenden Aktionen aus:

- Alle Prozesse, die in der WLM-Klasse ausgeführt werden, die ihren Grenzwert überschritten hat, werden beendet.
- Es wird nur der Prozess beendet, der für die Überschreitung des Grenzwerts der WLM-Klasse verantwortlich ist.
- Es wird der Prozess beendet, der seinen Prozessgrenzwert überschritten hat.

Es können virtuelle Speichergrenzen für jede benutzerdefinierte Klasse, jede Standardunterklasse in einer benutzerdefinierten Superklasse und die Standardsuperklasse angegeben werden.

Aus Abrechnungsgründen berücksichtigt WLM beim Ermitteln der WLM-Gesamtklassen- bzw. -Prozessauslastung nur Folgendes als virtuellen Speicher:

- Heap-Speicher
- Vom Ladeprogramm initialisierte Daten, BSS, gemeinsam genutzte Bibliotheken und privat geladene Segmente
- UBLOCK- und mmap-Bereiche
- Große und festgehaltene Seiten des Benutzeradressbereichs

Ein Administrator kann einen WLM-Grenzwert für den virtuellen Speicher für eine Klasse oder für jeden Prozess in der Klasse festlegen. Wenn ein Klassengrenzwert überschritten wird, kann WLM alle Prozesse, die der Klasse zugeordnet sind, oder nur den Prozess beenden, der für die Überschreitung des Grenzwerts verantwortlich ist. Dies richtet sich danach, ob das Klassenattribut **vmenforce** auf `class` oder `proc` gesetzt ist. Standardmäßig wird nur der Prozess beendet, der für die Überschreitung des Grenzwerts verantwortlich ist. Wenn ein Prozessgrenzwert definiert ist, wird ein Prozess beendet, wenn die virtuelle Speicherauslastung des Prozesses den Grenzwert überschreitet.

Betriebsarten für Workload Manager:

Mit WLM kann der Ressourcenverbrauch klassenbezogen auf prozentualer oder absoluter Basis oder prozessbezogen auf absoluter Basis reguliert werden. Die Regulierung für alle Ressourcentypen kann aktiviert werden, indem WLM im aktiven Modus ausgeführt wird.

Optional können Sie WLM in einem Modus starten, in dem neue und vorhandene Prozesse klassifiziert werden und die Ressourcenauslastung der verschiedenen Klassen überwacht wird, ohne die Auslastung zu regulieren. Dieser Modus ist der *passive Modus*.

Der passive Modus kann für die Konfiguration von WLM auf einem neuen System verwendet werden, um die Klassifizierungs- und Zuordnungsregeln zu prüfen und eine Vergleichsbasis für die Ressourcenauslastung für die verschiedenen Klassen zu erstellen, wenn die Prozessor- und Hauptspeicherzuordnung *nicht* von WLM reguliert wird. Dies gibt Systemadministratoren eine Basis, anhand derer sie entscheiden können, wie die Ressourcenanteile und Ressourcengrenzen (sofern erforderlich) angewendet werden müssen, um kritische Anwendungen zu bevorzugen und weniger wichtige Arbeit zu beschränken, damit die Geschäftsziele erreicht werden.

Wenn die Prozessorzeit die einzige Ressource ist, die Sie regulieren möchten, können Sie WLM im aktiven Modus für die Prozessorzeit und im passiven Modus für alle anderen Ressourcen ausführen. Der erste dieser beiden Modi wird als *reiner CPU-Modus* bezeichnet. Wenn Sie den Ressourcenverbrauch klassenbezogen auf prozentualer Basis regulieren und die Ressourcentypen auf absoluter Basis ausklammern möchten, können Abrechnung und Regulierung auf absoluter Basis für Klassen und Prozesse inaktiviert werden. In allen Modi haben Sie die Möglichkeit, die Ressourcengruppenbindung zu inaktivieren.

Dynamische Steuerung von Workload Manager:

Wenn WLM aktiv ist, kann jeder Parameter der aktuellen Konfiguration jederzeit geändert werden. Dazu gehören auch die Attribute einer Klasse, ihre Ressourcenanteile und -grenzwerte, die Zuordnungsregeln, das Hinzufügen neuer Klassen und das Löschen vorhandener Klassen.

Es stehen unter anderem die folgenden Möglichkeiten zur Verfügung:

- Sie können die Eigenschaftendateien für die derzeit aktive Konfiguration (Verzeichnis, auf das die symbolische Verbindung `/etc/wlm/current` zeigt) ändern und WLM mit dem Befehl **wlmcntrl** aktualisieren, so dass die neuen Parameter verwendet werden.
- Sie können eine weitere Konfiguration mit anderen Parametern erstellen und WLM anschließend aktualisieren, so dass die Parameter der neuen Konfiguration geladen werden und damit die neue Konfiguration zur aktuellen Konfiguration wird.

- Sie können einige Parameter der derzeit aktiven Konfiguration mit der WLM-Befehlszeilenschnittstelle (Befehle **mkclass**, **chclass** und **rmclass**) ändern.
- Sie können einige Parameter der derzeit aktiven Konfiguration mit einer Anwendung ändern, die die WLM-APIs verwendet.

Eine automatische Umstellung auf eine neue Konfiguration zu bestimmten Tageszeiten kann mit der Verwendung von *Konfigurationsgruppen* erreicht werden. Konfigurationsgruppen ermöglichen dem Administrator, eine Gruppe zu verwendender Konfigurationen und die Zeitbereiche anzugeben, in denen die einzelnen Konfigurationen jeweils aktiviert werden.

Überwachungstools:

Verwenden Sie die folgenden WLM-Befehle, um WLM-Statistiken anzuzeigen und den Betrieb von WLM zu überwachen.

- Der Befehl **wlmstat** ist textorientiert und zeigt Statistiken im Textformat an (prozentuale Ressourcenauslastung pro Klasse für alle Ressourcentypen, die von WLM verwaltet werden).
- Der Befehl **wlmmmon** gibt eine grafische Sicht der Ressourcenauslastung und WLM-Regulierung pro Klasse aus.
- Der Befehl **wlmpperf** ist ein optionales Tool, das in der Performance Toolbox verfügbar ist, und stellt weitere Funktionen bereit, z. B. langfristige Protokollierung und Wiedergabe.

Abwärtskompatibilität in Workload Manager:

In der Standardausgabe des Befehls **wlmstat** werden nur die Superklassen aufgelistet. Die Ausgabe gleicht der in den früheren Versionen.

Beispiel:

```
# wlmstat
      CLASS CPU MEM DKIO
Unclassified  0  0  0
  Unmanaged   0  0  0
    Default   0  0  0
    Shared    0  2  0
    System    2 12  0
    class1    0  0  0
    class2    0  0  0
#
```

Wenn ein WLM-Administrator für einige der Superklassen Unterklassen definiert hat, werden die Unterklassen aufgelistet. Beispiel:

```
# wlmstat
      CLASS CPU MEM DKIO
Unclassified  0  0  0
  Unmanaged   0  0  0
    Default   0  0  0
    Shared    0  2  0
    System    3 11  7
    class1   46  0  0
class1.Default 28  0  0
class1.Shared  0  0  0
  class1.sub1 18  0  0
    class2   48  0  0
#
```

Diese Ausgabe und die Ausgabe des Befehls **ps** sind identisch. Für Prozesse in einer Superklasse ohne Unterklassen listet der Befehl **ps** den Namen der Superklasse als Namen der Prozessklasse auf.

Klassenbezogene Abrechnung:

Mit dem AIX-Dienstprogramm für das Abrechnungssystem können Sie Informationen zur Verwendung verschiedener Systemressourcen nach Benutzer, Gruppe oder WLM-Klasse erfassen und Berichte erstellen.

Wenn die Prozessabrechnung aktiviert ist, zeichnet das Betriebssystem Statistiken über die Prozessressourcennutzung in einer Abrechnungsdatei auf, wenn der Prozess beendet wird. Dieser Abrechnungssatz enthält einen numerischen 64-Bit-Schlüssel, der den Namen der WLM-Klasse darstellt, zu der der Prozess gehört.

Das Abrechnungssystem verwendet einen 64-Bit-Schlüssel anstelle des vollständigen Klassennamens (34 Zeichen), um Speicherplatz einzusparen (andernfalls würde die Änderung die Größe des Abrechnungssatzes praktisch verdoppeln). Wenn der Abrechnungsbefehl ausgeführt wird, um die Benutzerprozessdaten zu extrahieren, wird der Schlüssel unter Verwendung der zuvor genannten Methode in den Klassennamen zurück übersetzt. Bei dieser Übersetzung werden die Klassennamen verwendet, die derzeit in den WLM-Konfigurationsdateien enthalten sind. Wenn eine Klasse also in dem Zeitraum zwischen Schreiben des Abrechnungssatzes bei Beendigung des Prozesses und Ausführung des Abrechnungsberichts gelöscht wurde, wird der Klassenname, der dem Schlüssel entspricht, nicht gefunden, und die Klasse wird als Unknown (Unbekannt) ausgewiesen.

Führen Sie einen der folgenden Schritte aus, um eine präzise Führung der Datensätze zur Ressourcennutzung von Klassen, die während einer Abrechnungsperiode gelöscht werden, sicherzustellen:

- Anstatt die Klasse zu löschen, behalten Sie den Klassennamen in der Datei classes und entfernen die Klasse aus der Datei rules, so dass der Klasse kein Prozess zugeordnet werden kann. In diesem Fall können Sie die Klasse löschen, nachdem der Abrechnungsbericht am Ende der Abrechnungsperiode generiert worden ist.
- Löschen Sie die Klasse aus der zugehörigen Konfiguration, und behalten Sie den Klassennamen in der Datei classes so lange in einer Pseudokonfiguration (einer Konfiguration, die niemals aktiviert wird), bis die Abrechnungssätze für den Zeitraum generiert wurden.

Zugehörige Konzepte:

„Systemabrechnung“ auf Seite 164

Mit dem Dienstprogramm für Systemabrechnung können Sie Informationen zur Verwendung verschiedener Systemressourcen für Einzelpersonen und Gruppen erfassen und zurückmelden.

Workload Manager verwalten

Mit Workload Manager (WLM) haben Systemadministratoren eine erheblich bessere Kontrolle darüber, wie der Scheduler und der Virtual Memory Manager (VMM) Ressourcen für Prozesse reservieren. Sie können mit WLM verhindern, dass unterschiedliche Klassen von Jobs miteinander in Konflikt geraten, und Sie können Ressourcen basierend auf den Anforderungen unterschiedlicher Benutzergruppen zuordnen.

Mit WLM können Sie unterschiedliche Serviceklassen für Jobs erstellen und Attribute für diese Klassen festlegen. Diese Attribute legen die Mindest- und Maximalwerte für CPU, physischen Speicher und Platten-E/A-Durchsatz für eine Klasse fest. WLM ordnet die Jobs anschließend unter Verwendung der Klassenzuordnungsregeln, die von einem Systemadministrator bereitgestellt werden, automatisch Klassen zu. Diese Zuordnungsregeln basieren auf den Werten einer Gruppe von Attributen für einen Prozess. Der Systemadministrator oder ein berechtigter Benutzer können Jobs auch manuell zu Klassen zuordnen und damit die automatische Zuordnung überschreiben.

WLM gehört zum Basisbetriebssystem und wird mit diesem installiert, ist aber ein optionaler Service. Der WLM-Service muss an die jeweilige Systemumgebung angepasst, bei Bedarf gestartet und gestoppt werden, wenn Sie ihn aussetzen oder beenden möchten.

Dieser Abschnitt enthält Prozeduren für die Konfiguration von WLM mit Klassen und Regeln, die für Ihren Standort geeignet sind, sowie Empfehlungen für die Fehlerbehebung bei unerwarteter Ressourcennutzung.

Achtung: In den Tasks in diesem Abschnitt wird davon ausgegangen, dass Sie mit WLM-Konzepten vertraut sind. Die effiziente Nutzung von WLM setzt umfassende Kenntnisse der vorhandenen Systemprozesse und der Systemleistung voraus. Wenn der Systemadministrator WLM mit extremen oder ungenauen Werten konfiguriert, hat dies eine erhebliche Verschlechterung der Leistung zur Folge.

Zugehörige Konzepte:

„Workload-Management-Konzepte“ auf Seite 515

Mit WLM können Sie unterschiedliche Serviceklassen für Jobs erstellen und Attribute für diese Klassen festlegen.

Workload Manager starten und stoppen:

WLM ist ein optionaler Service, der gestartet und gestoppt werden muss.

Es wird empfohlen, eine der Systemverwaltungsschnittstellen oder SMIT zu verwenden, um WLM zu starten und zu stoppen.

- Verwenden Sie zum Starten oder Stoppen von WLM mit SMIT den Direktaufruf `smi t wlmmanage`.

Der Hauptunterschied zwischen diesen Optionen ist die Permanenz. In SMIT können Sie WLM auf drei Arten starten und stoppen:

Sofort Wenn Sie WLM mit dieser Option stoppen, wird WLM nur für diese Sitzung gestoppt und beim nächsten Warmstart erneut gestartet. Wenn Sie WLM mit dieser Option starten, wird WLM nur für diese Sitzung gestartet und beim nächsten Warmstart nicht erneut gestartet.

Beim nächsten Warmstart

Wenn Sie WLM mit dieser Option stoppen, bleibt WLM nur für diese Sitzung aktiv und wird beim nächsten Warmstart *nicht* erneut gestartet. Wenn Sie WLM mit dieser Option starten, ist WLM für diese Sitzung nicht verfügbar, wird aber beim nächsten Warmstart gestartet.

Beides

Wenn Sie WLM mit dieser Option stoppen, wird WLM nur für diese Sitzung gestoppt und beim nächsten Warmstart *nicht* erneut gestartet. Wenn Sie WLM mit dieser Option starten, wird WLM nur für diese Sitzung gestartet *und* beim nächsten Warmstart erneut gestartet.

Sie können auch den Befehl `wlmcntrl` verwenden, aber mit dem Befehl `wlmcntrl` können Sie WLM nur für die aktuelle Sitzung starten und stoppen. Wenn Sie die Befehlszeilenschnittstelle verwenden möchten und die Änderung einen Warmstart des Systems überdauern soll, müssen Sie die Datei `/etc/inittab` ändern.

Mit WLM kann der Ressourcenverbrauch klassenbezogen auf prozentualer oder absoluter Basis oder prozessbezogen auf absoluter Basis reguliert werden. Die Regulierung für alle Ressourcentypen kann aktiviert werden, indem WLM im *aktiven* Modus ausgeführt wird. Optional können Sie WLM in einem Modus starten, in dem neue und vorhandene Prozesse klassifiziert werden und die Ressourcenauslastung der verschiedenen Klassen überwacht wird, ohne die Auslastung zu regulieren. Dieser Modus ist der *passive* Modus. Wenn die CPU-Zeit die einzige Ressource ist, die Sie regulieren möchten, können Sie WLM im aktiven Modus für die CPU-Zeit und im passiven Modus für alle anderen Ressourcen ausführen. Der erste dieser beiden Modi wird als *reiner CPU-Modus* bezeichnet.

Alle Prozesse, die beim Start von WLM bereits im System vorhanden sind, werden entsprechend den neu geladenen Zuordnungsregeln klassifiziert und von WLM überwacht.

Eigenschaften von Workload Manager:

Sie können die Eigenschaften für die WLM-Konfiguration mit SMIT, der WLM-Befehlszeilenschnittstelle oder durch Erstellen von unstrukturierten ASCII-Dateien angeben. Die SMIT-Schnittstellen verwenden die WLM-Befehle, um die Informationen in denselben unstrukturierten ASCII-Dateien, so genannten *Eigenschaftendateien*, aufzuzeichnen.

Eine WLM-Definition wird mit einer Gruppe von WLM-Eigenschaftendateien definiert. Sie können mehrere Gruppen von Eigenschaftendateien erstellen, die jeweils eine andere Konfiguration des Workload-Managements definieren. Diese Konfigurationen werden in Unterverzeichnissen des Verzeichnisses */etc/wlm* gespeichert. Bei den WLM-Eigenschaftendateien, die die Superklassen der Konfiguration *Config* beschreiben, handelt es sich um die Dateien *classes*, *description*, *limits*, *shares* und *rules* im Verzeichnis */etc/wlm/Config*. Die Eigenschaftendateien, die die Unterklassen der Superklasse *Super* dieser Konfiguration beschreiben, sind *classes*, *limits*, *shares* und *rules* im Verzeichnis */etc/wlm/Config/Super*. Nur Root kann WLM starten und stoppen und eine andere Konfiguration auswählen.

Die Eigenschaftendateien haben die folgenden Namen:

| Eintrag | Beschreibung |
|--------------------|-------------------------------------|
| classes | Klassendefinitionen |
| description | Beschreibungstext für Konfiguration |
| groupings | Gruppierungen von Attributwerten |
| limits | Grenzwerte für Klassen |
| shares | Zielanteile für Klassen |
| rules | Regeln für die Klassenzuordnung |

Mit dem Befehl **wlmcntrl**, mit dem die WLM-Eigenschaftendateien übergeben werden, und den anderen WLM-Befehlen können Benutzer einen anderen Verzeichnisnamen für die WLM-Eigenschaftendateien angeben. Auf diese Weise können Sie die WLM-Eigenschaften ändern, ohne die WLM-Standard-eigenschaftendateien zu ändern.

Die symbolische Verbindung */etc/wlm/current* zeigt auf das Verzeichnis, das die aktuellen Konfigurationsdateien enthält. Sie können diese Verbindung mit dem Befehl **wlmcntrl** aktualisieren, wenn Sie WLM mit einer bestimmten Konfiguration oder Konfigurationsgruppe starten. Die Beispielfunktionsdateien, die mit dem Betriebssystem geliefert werden, sind im Verzeichnis */etc/wlm/standard* gespeichert.

Eine Attributwertgruppe erstellen:

Sie können Attributwerte gruppieren und sie mit einem einzigen Wert in der Datei *rules* darstellen. Diese *Attributwertgruppierungen* werden in der Datei *groupings* im WLM-Konfigurationsverzeichnis definiert.

Standardmäßig hat eine Konfiguration keine Datei *groupings*. Zum Erstellen einer solchen Datei ist weder ein Befehl noch eine Verwaltungsschnittstelle verfügbar. Sie müssen die folgende Prozedur verwenden, um Attributwertgruppierungen zu erstellen und zu verwenden:

1. Melden Sie sich als Root an, und wechseln Sie, wie im folgenden Beispiel gezeigt, in das entsprechende Konfigurationsverzeichnis:
`cd /etc/wlm/MyConfig`
2. Erstellen und bearbeiten Sie in dem von Ihnen bevorzugten Editor eine Datei mit dem Namen *groupings*. Beispiel:
`vi groupings`
3. Definieren Sie Attribute und die zugehörigen Werte in dem folgenden Format:

Attribut = Wert, Wert, ...

Alle Werte müssen durch Kommas voneinander getrennt werden. Leerzeichen werden nicht berücksichtigt. Es können auch Bereiche und Platzhalterzeichen angegeben werden. Beispiel:

```
trusted = user[0-9][0-9], admin*
nottrusted = user23, user45
shell = /bin/?sh, \
        /bin/sh, \
        /bin/tcsh
rootgroup=system,bin,sys,security,cron,audit
```

4. Speichern Sie die Datei.
5. Wenn Sie Attributgruppierungen in den Auswahlkriterien für eine Klasse verwenden möchten, editieren Sie die Datei `rules`. Dem Namen der Attributgruppierung muss ein Dollarzeichen (\$) vorangestellt werden. Wenn Sie die Werte der Attributgruppierung einschließen möchten, geben Sie `$Name_der_Attributgruppierung` an. Wenn Sie die Werte der Attributgruppierung ausschließen möchten, verwenden Sie ein Ausrufezeichen (!), gefolgt von `$Name_der_Attributgruppierung`. Das Ausrufezeichen kann nicht in den Einträgen der Gruppe verwendet werden (Schritt 3 auf Seite 524) und ist der einzige Modifikator, der der Gruppierung in dieser Datei `rules` vorangestellt werden kann. Im folgenden Beispiel kennzeichnet der Stern (*) eine Kommentarzeile:

```
*class resvd user group application type tag
classA - $trusted,!$nottrusted - - - -
classB - - - $shell,!/bin/zsh - -
classC - - $rootgroup - - -
```

6. Speichern Sie die Datei.

Jetzt enthalten Ihre Klassifizierungsregeln Attributwertgruppierungen. Wenn bei der Auswertung der Regeln ein Element mit einem Dollarzeichen (\$) beginnt, sucht das System nach diesem Element in der Datei `groupings`. Ist ein Element syntaktisch ungültig oder ist die Datei `groupings` nicht vorhanden, zeigt das System eine Warnung an und fährt mit der Verarbeitung der anderen Regeln fort.

Eine zeitbasierte Konfigurationsgruppe erstellen:

Sie können eine Gruppe spezieller Konfigurationen erstellen und jede Konfiguration in der Gruppe Tagen und Zeiten zuordnen, an denen die jeweilige Konfiguration aktiviert werden soll.

Diese Gruppen, die so genannten *Konfigurationsgruppen*, sind vollständig unabhängig von, aber kompatibel mit Ihrer normalen Konfiguration. Mit dem Befehl `wlmcntrl -u` können Sie bei Bedarf zwischen einer Konfigurationsgruppe und Ihrer normalen Konfiguration wechseln.

Wenn Sie eine Konfigurationsgruppe verwenden, ordnen Sie vorhandene benannte Konfigurationen gewöhnlich einem bestimmten Zeitbereich zu. Da jeweils nur eine Konfiguration verwendet werden kann, muss jeder angegebene Zeitbereich eindeutig sein. Zeitbereiche dürfen weder überlappen noch doppelt vorkommen.

Der Dämon `wlmd` warnt WLM, wenn der Zeitbereich für eine bestimmte Konfiguration abläuft und eine andere Konfiguration verwendet werden muss. Diese Zeitbereiche, die im Verzeichnis der Konfigurationsgruppe in einer ASCII-Datei mit dem Namen `.times` angegeben werden, können nur von Root verwaltet werden.

Verwenden Sie die folgende Prozedur, um eine zeitbasierte Konfigurationsgruppe zu erstellen:

1. Melden Sie sich als Benutzer mit Rootberechtigung an. Erstellen Sie ein Verzeichnis für die Konfigurationsgruppe, und wechseln Sie anschließend in dieses Verzeichnis. Beispiel:

```
mkdir /etc/wlm/MyConfigSet
cd /etc/wlm/MyConfigSet
```

2. Rufen Sie den von Ihnen bevorzugten Editor auf, und erstellen Sie die Datei `.times` für die Konfigurationsgruppe. Geben Sie in dieser Datei die Konfiguration und die Zeitbereiche im folgenden Format an:

```
Konfigurationsname:  
    time = "N-N,HH:MM-HH:MM"
```

oder

```
Konfigurationsname:  
    time = -
```

(Kein Zeitwert angegeben) *N* steht für eine Nummer, die einen Tag der Woche (Bereich 0 (Sonntag) bis 6 (Samstag)) angibt, *HH* für die Stunde (Bereich 00 (Mitternacht) bis 23 (11 Uhr abends)) und *MM* für die Minuten (Bereich 00 bis 59). Sie können auch nur einen Tag oder gar keinen Tag angeben. Der Stundenwert 24 kann für die letzte Stunde des Tages angegeben werden, sofern als Minutenwert 00 angegeben wird. Wenn Sie ein Minuszeichen (-) anstelle eines Zeitbereichs für eine bestimmte Konfiguration angeben, wird diese Konfiguration verwendet, wenn die Zeitbereiche der anderen Konfigurationen nicht gültig sind. Es kann nur eine Konfiguration ohne Zeitbereich angegeben werden.

Beispiel:

```
conf1:  
    time =  
conf2:  
    time = "1-5,8:00-17:00"  
conf2  
    time = "6-0,14:00-17:00"  
conf3  
    time = "22:00-6:00"
```

3. Verwenden Sie den Befehl **wlmcntrl -u**, um WLM mit der neuen Konfigurationsgruppe zu aktualisieren. Beispiel:

```
wlmcntrl -u /etc/wlm/MyConfigSet
```

Jetzt ist die aktuelle WLM-Konfiguration Ihre neue zeitbasierte Konfigurationsgruppe.

Sie können auch die Befehle **confsetcntrl** und **lswlmconf** verwenden, um Konfigurationsgruppen zu erstellen und zu bearbeiten. Beispiel:

Verwenden Sie den folgenden Befehl, um die Konfigurationsgruppe **confset1** mit der Standardkonfiguration **conf1** zu erstellen:

```
confsetcntrl -C confset1 conf1
```

Verwenden Sie den folgenden Befehl, um **conf2** zu **confset1** hinzuzufügen und täglich in der Zeit von 8:00 Uhr morgens bis 5:00 nachmittags als aktive Konfiguration festzulegen:

```
confsetcntrl -d confset1 -a conf2 "0-6,08:00-17:00"
```

Verwenden Sie den folgenden Befehl, um diese Konfigurationsgruppe als aktive Konfiguration festzulegen:

```
wlmcntrl -d confset1
```

Eine Ressourcengruppe erstellen:

Die Verwendung von Ressourcengruppen (rsets, Resource Sets) ist eine effektive Methode, um Workloads voneinander abzugrenzen, soweit es die CPU betrifft. Indem Sie zwei unterschiedliche Workloads in zwei Klassen trennen und jeder Klasse eine andere Untergruppe der CPUs zuteilen, können Sie sicherstellen, dass die beiden Workloads nicht um CPU-Ressourcen konkurrieren. Eine Konkurrenzsituation um physischen Speicher und E/A-Bandbreite ist jedoch damit nicht ausgeschlossen.

Am einfachsten lässt sich eine Ressourcengruppe mit der Schnittstelle SMIT (Direktauftrag **smit addrsetcntrl**) oder mit dem Befehl **mkrset** erstellen.

Zur Anleitung veranschaulicht das folgende Beispiel die einzelnen Schritte, die zum Erstellen und Benennen einer Ressourcengruppe auf einem 4-Wege-System ausgeführt werden müssen. In diesem Beispiel soll eine Ressourcengruppe mit den Prozessen 0 bis 2 erstellt und dann in der WLM-Konfiguration verwendet werden, um alle Prozesse auf eine Superklasse dieser drei Prozessoren zu beschränken.

1. Melden Sie sich als Root an, und zeigen Sie mit dem folgenden Befehl die verfügbaren Bausteine (aus denen Sie die Ressourcengruppen erstellen) an:

```
lsrset -av
```

Die Ausgabe dieses Beispielbefehls ist wie folgt:

| T | Name | Owner | Group | Mode | CPU | Memory | Resources |
|---|----------------|-------|--------|--------|-----|--------|---------------|
| r | sys/sys0 | root | system | r----- | 4 | 98298 | sys/sys0 |
| r | sys/node.00000 | root | system | r----- | 4 | 98298 | sys/sys0 |
| r | sys/mem.00000 | root | system | r----- | 0 | 98298 | sys/mem.00000 |
| r | sys/cpu.00003 | root | system | r----- | 1 | 0 | sys/cpu.00003 |
| r | sys/cpu.00002 | root | system | r----- | 1 | 0 | sys/cpu.00002 |
| r | sys/cpu.00001 | root | system | r----- | 1 | 0 | sys/cpu.00001 |
| r | sys/cpu.00000 | root | system | r----- | 1 | 0 | sys/cpu.00000 |

In der Ausgabe stellt **sys/sys0** das gesamte System dar (in diesem Fall ein 4-Wege-SMP-System). Besitzt eine WLM-Klasse nicht das Attribut **rset**, ist dies die Standardgruppe, auf die die Prozesse der Klasse zugreifen können.

2. Erstellen und benennen Sie mit dem folgenden SMIT-Direktaufruf die Ressourcengruppe:

```
smit addrsetcntl
```

Füllen Sie für dieses Beispiel die Felder folgendermaßen aus:

Namespace

admin

Name der Ressourcengruppe

proc0_2

Ressourcen

Wählen Sie in der Liste die Zeilen aus, die dem Hauptspeicher und den CPUs 0 bis 2 (sys/cpu.00000 bis sys.cpu.00002) entsprechen.

Alle anderen Felder.

Wählen Sie die gewünschten Einträge aus den Listen aus.

Nachdem Sie die Felder ausgefüllt und SMIT beendet haben, ist die Ressourcengruppe admin/proc0_2 in /etc/rsets erstellt.

3. Wenn Sie die neue Ressourcengruppe verwenden möchten, fügen Sie sie mit dem folgenden SMIT-Direktaufruf den Kerneldatenstrukturen hinzu:

```
smit reloadrsetcntl
```

Dieses Menü enthält Optionen, über die Sie die Datenbank sofort, beim nächsten Bootvorgang oder sofort und beim nächsten Bootvorgang erneut laden können. Da dies das erste Mal ist, dass Sie die neue Ressourcengruppe verwenden, wählen Sie die Option Beides (sofort und beim nächsten Bootvorgang) aus, damit die Ressourcengruppe sofort und nach jedem Bootvorgang geladen wird. (Falls Sie eine vorhandene Ressourcengruppe ändern, sollten Sie die Option sofort auswählen.)

4. Fügen Sie die neue Ressourcengruppe mit dem folgenden SMIT-Direktaufruf einer WLM-Klasse hinzu:

```
smit wlmclass_gal
```

Wählen Sie die Klasse (in diesem Beispiel **super1**) und anschließend in der im Feld **Ressourcengruppe** verfügbaren Liste den Eintrag **admin/proc0_2** aus., Nachdem Sie Ihre Auswahl getroffen und SMIT beendet haben, ist die Datei classes auf der Platte geändert.

5. Führen Sie eine der folgenden Aktionen aus:

- Wenn WLM aktiv ist, aktualisieren Sie die Konfiguration mit dem folgenden SMIT-Direktaufruf:
smit wlmupdate
- Wenn WLM nicht aktiv ist, starten Sie ihn mit dem folgenden SMIT-Direktaufruf:
smit wlmstart

6. Überwachen Sie die Auswirkungen auf die neue Ressourcengruppe in der Klasse. Beispiel:

- a. Starten Sie 90 CPU-Schleifen (Programmausführung in einer Endlosschleife) in der Klasse **super1**.
- b. Geben Sie in der Befehlszeile wlmstat ein. Die Ausgabe dieses Beispielbefehls ist wie folgt:

```
CLASS CPU MEM B10
Unclassified 0 0 0
Unmanaged 0 0 0
Default 8 0 0
Shared 0 0 0
System 0 0 0
super1 75 0 0
super2 0 0 0
super2.Default 0 0 0
super2.Shared 0 0 0
super2.sub1 0 0 0
super2.sub2 0 0 0
```

Diese Ausgabe zeigt, dass die 90 CPU-gebundenen Prozesse, die ansonsten uneingeschränkt 100 % der CPU-Leistung in Anspruch nehmen würden, jetzt nur 75 % nutzen, da die Ressourcengruppe sie auf die Ausführung in den CPUs 0 bis 2 beschränkt.

- c. Verwenden Sie den folgenden SMIT-Direktaufruf, um zu prüfen, auf welche Ressourcengruppe ein Prozess (angegeben mit seiner PID) Zugriff hat:

```
smit lsrsetproc
```

Geben Sie die PID des Prozesses ein, an dem Sie interessiert sind, oder wählen Sie sie in der Liste aus. Die folgende Ausgabe bezieht sich auf einen der in der Schleife ausgeführten Prozesse:

```
CPU Memory Resources
3 98298 sys/mem.00000 sys/cpu.00002 sys/cpu.00001 sys/cpu.00000
```

Ein Prozess aus einer Klasse ohne spezifiziertes **rset**-Attribut verwendet jedoch die Standardressourcengruppe (**Default**), der alle Prozessoren mit Ausnahme derer enthält, die zu einer exklusiven Ressourcengruppe gehören. Ein Prozess, der zu keiner Klasse gehört, verwendet die Klasse **System** (sofern es sich um einen Stammprozess handelt) oder eine Standardklasse (**Default**) (sofern es sich um einen Prozess handelt, der kein Stammprozess ist). Für jede dieser Klassen können Ressourcengruppen definiert sein.

Die folgende Ausgabe stammt vom Prozess **init**, der sich in einer Klasse befindet, für die keine Ressourcengruppe definiert ist:

```
CPU Memory Resources
4 98298 sys/sys0
```

Jetzt ist Ihre Ressourcengruppe vorhanden und wird von mindestens einer Klasse in WLM verwendet.

Anmerkung: WLM verwendet seine rset-Zuordnung nicht für Prozesse, die derzeit über die Subroutine **bindprocessor** oder über eine andere rset-Zuordnung gebunden sind. Wenn die andere Zuordnung nicht mehr vorhanden ist, ordnet WLM automatisch seine Ressourcengruppe zu.

Anmerkung: Ressourcengruppen können für jede WLM-Klasse erstellt werden, die die Klassen **Default** und **System** enthält.

Zugehörige Konzepte:

„Registry für Ressourcengruppen in Workload Manager“ auf Seite 552

Mit den Registry-Services für Ressourcengruppen (**rset**) können Systemadministratoren Ressourcengruppen definieren und benennen, so dass sie von anderen Benutzern oder Anwendungen verwendet werden können.

Zugehörige Informationen:

lsrset command

Workload Manager für die Konsolidierung von Workloads konfigurieren:

Mit Workload Manager (WLM) können Sie steuern, welche Ressourcen von Jobs auf Ihrem System verwendet werden.

Unter jedem installierten Betriebssystem AIX gibt es eine WLM-Standardkonfigurationsschablone. Die folgende Prozedur aktualisiert die WLM-Konfigurationsschablone so, dass sie eine Richtlinie für Ressourcenverwaltung auf einem gemeinsam benutzten Server implementiert. Die sich daraus ergebende Konfiguration kann als Ausgangspunkt für Tests verwendet werden. Wie Sie den WLM genau konfigurieren müssen, hängt von der Auslastung und den Anforderungen Ihrer Umgebung ab.

Anmerkung:

1. Die effiziente Nutzung von WLM setzt umfassende Kenntnisse der vorhandenen Systemprozesse und der Systemleistung voraus. Sie werden wahrscheinlich wiederholt Tests und Optimierungsschritte ausführen müssen, bis Sie eine an die gegebene Auslastung angepasste Konfiguration erstellt haben. Wenn Sie den WLM mit extremen oder ungenauen Werten konfigurieren, kann der Systemdurchsatz erheblich zurückgehen.
2. Wenn Sie bereits einige Klassifizierungsattribute eines Prozesses kennen (z. B. den Benutzer-, Gruppen- oder Anwendungsnamen), vereinfacht dies die Konfiguration des WLM. Sollten Sie mit der aktuellen Ressourcenverwendung nicht vertraut sein, verwenden Sie ein Tool wie **topas**, um festzustellen, welche Prozesse die Hauptressourcennutzer sind. Die Ergebnisse des Tools können Sie dann als Grundlage für das Definieren von Klassen und Regeln verwenden.
3. Das folgende Szenario setzt voraus, dass Sie mit den grundlegenden Konzepten von Workload Manager vertraut sind. Eine Beschreibung dieser Konzepte finden Sie im Abschnitt „Workload-Management-Konzepte“ auf Seite 515.

Die WLM-Konfigurationsdateien befinden sich im Verzeichnis `/etc/wlm/Konfigurationsname`. Die Unterklassen aller Superklassen sind in einer Konfigurationsdatei mit dem Namen `/etc/wlm/Konfigurationsname/Superklassenname` definiert. Weitere Informationen zu diesen Dateien finden Sie in der Veröffentlichung *Files Reference*.

Mit der folgenden Prozedur werden Sie die Arbeitsbelastung von zwei verschiedenen Abteilungsservern auf einem großen Server konsolidieren. In diesem Beispiel werden die Konfigurationsdateien bearbeitet, aber Sie können eine Konfiguration auch mit SMIT erstellen (verwenden Sie den Direktaufruf **smit wlm-config_create**). Diese Prozedur umfasst im Wesentlichen die folgenden Schritte:

1. Bestimmen Sie den Ressourcenbedarf der zu konsolidierenden Anwendungen. So können Sie besser abschätzen, wie viele Anwendungen Sie auf den großen Server verschieben können.
2. Definieren Sie Schichten, Ressourcenanteile und Ressourcengrenzen, um die konsolidierte Auslastung testen zu können.
3. Optimieren Sie die Konfiguration, bis sich die gewünschten Ergebnisse einstellen.

Diese Verwendungshinweise wurden unter verschiedenen AIX-Versionen getestet. Die Ergebnisse, die Sie erhalten, können je nach Versionsstand des Betriebssystems AIX signifikant abweichen.

Schritt 1. Anwendungsanforderungen ermitteln

In diesem Szenario wird von der für einen Datenbankserver typischen Arbeitsbelastung ausgegangen. Die Jobs lassen sich den folgenden allgemeinen Kategorien zuordnen:

Listener

Dies sind Prozesse, die die meiste Zeit inaktiv sind und immer dann aktiviert werden, wenn Sie auf eine Anfrage reagieren müssen. Diese Prozesse benötigen nur geringe Ressourcen. Die Antwortzeit kann dennoch kritisch sein.

Worker

Dies sind Prozesse, die die mit einer lokalen oder fernen Anfrage verbundene Arbeit ausführen. Sie werden deshalb wahrscheinlich viel CPU-Zeit und Speicher benötigen.

Reporter

Dies sind Prozesse, die automatische Tasks ausführen. Sie können viel CPU-Zeit und Speicher benötigen, aber auch eine längere Antwortzeit tolerieren.

Monitore

Diese Prozesse laufen regelmäßig ab, um den Status des Systems oder von Anwendungen zu prüfen. Sie können eine beachtliche Menge an Ressourcen benötigen, allerdings nur für kurze Zeit.

Befehle

Dies sind Befehle oder andere Anwendungen, die jederzeit von Systembenutzern ausgeführt werden können. Der Ressourcenbedarf ist daher nicht vorhersagbar.

Darüber hinaus können die Jobs wie folgt eingestuft werden:

SysTools

Dies sind Prozesse, die automatische Tasks ausführen. Diese Jobs sind für den Systembetrieb nicht kritisch, müssen jedoch regelmäßig innerhalb bestimmter zeitlicher Vorgaben ausgeführt werden.

SysBatch

Diese unregelmäßig ausgeführten Prozesse sind nicht kritisch für den Systembetrieb und an keine zeitlichen Vorgaben gebunden.

Der erste Schritt für die Erstellung einer Konfiguration ist das Definieren von Klassen und Regeln. Im Folgenden werden Sie beim Definieren der Klassen die oben aufgelisteten allgemeinen Jobkategorien verwenden. Gehen Sie wie folgt vor:

1. Erstellen Sie mit dem folgenden Befehl im Verzeichnis `/etc/wlm` die neue Konfiguration `MyConfig`:
`mkdir /etc/wlm/MyConfig`
2. Kopieren Sie mit dem folgenden Befehl die Schablondateien in das Verzeichnis `/etc/wlm/MyConfig`:
`cp -pr /etc/wlm/template/* /etc/wlm/MyConfig`
3. Zum Erstellen der Superklassen ändern Sie in dem von Ihnen bevorzugten Editor die Datei `/etc/wlm/MyConfig/classes` so, dass sie Folgendes enthält:

System:

Default:

DeptA:

DeptB:

SysTools:

SysBatch:

Definieren Sie zunächst für jede Abteilung eine Superklasse (denn der Server wird von zwei Abteilungen gemeinsam benutzt). Die Superklassen `SysTool` und `SysBatch` werden die geplanten Jobs bearbeiten, die oben in den allgemeinen Kategorien beschrieben sind. Die Superklassen `System` und `Default` sind immer definiert.

4. Erstellen Sie im Verzeichnis `MyConfig` für jede der Superklassen `DeptA` und `DeptB` ein Verzeichnis. (Wenn Sie eine Konfiguration erstellen, müssen Sie für jede Superklasse mit Unterklassen ein geson-

iertes Verzeichnis erstellen.) Im folgenden Schritt werden Sie zur Superklasse jeder Abteilung fünf Unterklassen definieren (für jeden Vorgangstyp eine).

5. Zum Erstellen von Unterklassen für jede allgemeine Jobkategorie müssen Sie die Dateien `/etc/wlm/MyConfig/DeptA/classes` und `/etc/wlm/MyConfig/DeptB/classes` editieren, so dass sie Folgendes enthalten:

Listen:

Work:

Monitor:

Report:

Command:

Anmerkung: Der Inhalt der Datei `classes` kann für jede Superklasse anders sein.

Nachdem Sie die Klassen definiert haben, werden Sie im folgenden Schritt die Klassifizierungsregeln für die Klassifizierung von Prozessen auf Superklassen- und Unterklassenebene erstellen. Zur Vereinfachung wollen wir davon ausgehen, dass der Ausführungsort aller Anwendungen bekannt ist. Alle Prozesse der einen Abteilung werden unter der UNIX-Gruppe `deptA` und die der anderen Abteilung unter der UNIX-Gruppe `deptB` ausgeführt.

6. Zum Erstellen der Zuordnungsregeln für die Superklasse müssen Sie die Datei `/etc/wlm/MyConfig/rules` so modifizieren, dass Sie Folgendes enthält:

```
DeptA - - deptA - -  
DeptB - - deptB - -  
SysTools - root,bin - /usr/sbin/tools/* -  
SysBatch - root,bin - /usr/sbin/batch/* -  
System - root - - -  
Default - - - - -
```

Anmerkung: Wenn mehr als eine Instanz einer Anwendung ausgeführt werden kann und alle Klassifizierungsattribute (mit Ausnahme des Tags) identisch sind, können Sie den Befehl `wlmassign` oder die Subroutine `wlm_set_tag` verwenden, um die Instanzen zur besseren Unterscheidung verschiedenen Klassen zuzuordnen.

7. Wenn Sie spezifischere Unterklassenregeln verwenden möchten, erstellen Sie die Dateien `/etc/wlm/MyConfig/DeptA/rules` und `/etc/wlm/MyConfig/DeptB/rules` mit folgendem Inhalt:

```
Listen - - - /opt/myapp/bin/listen* -  
Work - - - /opt/myapp/bin/work* -  
Monitor - - - /opt/bin/myapp/bin/monitor -  
Report - - - /opt/bin/myapp/report* -  
Command - - - /opt/commands/* -
```

8. Bestimmen Sie den Ressourcenbedarf der einzelnen Klassen. Starten Sie dazu den WLM mit folgendem Befehl im passiven Modus:

```
wlmcntrl -p -d MyConfig
```

Anschließend können Sie zunächst jede Anwendung einzeln ausführen, um einen genaueren Überblick über die jeweils benötigten Ressourcen zu erhalten. Führen Sie dann alle Anwendungen parallel aus. So können Sie die Interaktion aller Klassen besser beurteilen.

Eine alternative Methode zur Bestimmung des Ressourcenbedarfs wäre die Ausführung des WLM im passiven Modus auf jedem der beiden Server, deren Arbeitsbelastung Sie konsolidieren möchten. Der Nachteil dieser Methode ist, dass Sie die Konfigurationen auf dem größeren System erneut erstellen müssten und dass der benötigte Ressourcenanteil auf dem größeren System wahrscheinlich anders aussehen würde.

Schritt 2. Schichten, Anteile und Grenzwerte definieren

Eine WLM-Konfiguration ist die Implementierung einer Richtlinie für Ressourcenverwaltung. Wenn Sie WLM im passiven Modus ausführen, erhalten Sie Informationen, anhand derer Sie einschätzen können, ob Ihre Richtlinie für Ressourcenverwaltung für die gegebene Arbeitsbelastung angemessen ist. Sie können Schichten, Anteile und Grenzwerte definieren, um die Arbeitsbelastung entsprechend Ihrer Richtlinie für Ressourcenverwaltung zu regulieren.

Für dieses Szenario werden die folgenden Anforderungen vorausgesetzt:

- Die Klasse System muss die höchste Priorität haben. Ihr muss jederzeit der Zugriff auf einen bestimmten Teil der Systemressourcen gewährt werden.
- Die Klasse SysTools muss ständig Zugriff auf einen bestimmten Teil der Ressourcen haben. Ihr Ressourcenbedarf ist allerdings so niedrig, dass er kaum negative Auswirkungen auf die in den Abteilungen DeptA und DeptB ausgeführten Anwendungen haben wird.
- Die Klasse SysBatch kann die übrigen Arbeitsprozesse des Systems nicht stören.
- DeptA werden 60 % der verfügbaren Ressourcen (d. h. der für Klassen mit Anteilen verfügbaren Ressourcen) zugeordnet und DeptB 40 %. Innerhalb der Abteilungen DeptA und DeptB gilt Folgendes:
 - Prozesse in der Klasse Listen müssen mit geringer Latenzzeit auf Anfragen reagieren, dürfen aber nicht viele Ressourcen nutzen.
 - Für die Klasse Work müssen die meisten Ressourcen bereitgestellt werden. Die Klassen Monitor und Command benötigen ebenfalls Ressourcen, jedoch weniger als die Klasse Work.
 - Die Klasse Report kann die übrigen Arbeitsprozesse nicht stören.

Mit der folgenden Prozedur definieren Sie Schichten, Anteile und Grenzwerte:

1. Zum Erstellen der Superklassenschichten ändern Sie in dem von Ihnen bevorzugten Editor die Datei `/etc/wlm/MyConfig/classes` so, dass sie Folgendes enthält:

```
System:
```

```
Default:
```

```
DeptA:
```

```
    localshm = yes  
    adminuser = adminA  
    authuser = adminA  
    inheritance = yes
```

```
DeptB:
```

```
    localshm = yes  
    adminuser = adminB  
    authuser = adminB  
    inheritance = yes
```

```
SysTools:
```

```
    localshm = yes
```

```
SysBatch:
```

```
    tier = 1  
    localshm = yes
```

Die Superklasse SysBatch wird in die Schicht 1 gestellt, weil diese Klasse nur Jobs mit sehr niedriger Priorität enthält, die die übrigen Arbeitsprozesse des Systems nicht beeinträchtigen sollen. (Wenn keine Schicht angegeben ist, wird die Klasse standardmäßig Schicht 0 zugeordnet.) Die Verwaltung der Superklasse für jede Abteilung wird durch die Attribute `adminuser` und `authuser` definiert. Für DeptA und DeptB ist das Attribut `inheritance` aktiviert. Alle neuen Prozesse, die in einer Klasse mit "inheritance" gestartet werden, bleiben dieser Klasse zugeordnet.

2. Zum Erstellen von Unterklassenschichten für jede Jobgruppe müssen Sie die Dateien `/etc/wlm/MyConfig/DeptA/classes` und `/etc/wlm/MyConfig/DeptB/classes` so modifizieren, dass sie Folgendes enthalten:

Listen:

Work:

Monitor:

Report:

```
tier = 1
```

Command:

3. Ordnen Sie den Superklassen die ersten Anteile zu. Dazu müssen Sie die Datei `/etc/wlm/MyConfig/shares` editieren, so dass sie Folgendes enthält:

DeptA:

```
CPU = 3  
memory = 3
```

DeptB:

```
CPU = 2  
memory = 2
```

Sie haben insgesamt fünf CPU-Anteile zugeordnet. DeptA-Prozesse haben demzufolge Zugriff auf drei der fünf Anteile (oder 60 % der gesamten CPU-Ressourcen) und DeptB-Prozesse auf zwei der fünf Anteile (oder 40 %). Da Sie den Klassen SysTools, System und Default keine Anteile zugeordnet haben, ist der Ressourcenverbrauch dieser Klassen von der Anzahl der aktiven Anteile unabhängig. Der Ressourcenzugriff dieser Klassen hat somit Vorrang vor dem Ressourcenzugriff von DeptA und DeptB (bis für die beiden letztgenannten der Ressourcengrenzwert erreicht ist). Der Klasse SysBatch wurden keine Anteile zugeordnet, weil sie die einzige Superklasse in Schicht 1 ist, so dass Zuordnungen von Anteilen irrelevant sind. Jobs in der Klasse SysBatch können nur die Ressourcen nutzen, die von allen anderen Klassen in Schicht 0 nicht benutzt werden.

4. Ordnen Sie den Unterklassen die ersten Anteile zu. Dazu müssen Sie die Dateien `/etc/wlm/MyConfig/DeptA/shares` und `/etc/wlm/MyConfig/DeptB/shares` editieren, so dass sie Folgendes enthalten:

Work:

```
CPU = 5  
memory = 5
```

Monitor:

```
CPU = 4  
memory = 1
```

Command:

```
CPU = 1  
memory = 1
```

Da Sie der Klasse Listen keine Anteile zugeordnet haben, hat der bedarfsabhängige Ressourcenzugriff dieser Klasse die höchste Priorität (in der Superklasse). Die meisten Anteile wurden der Klasse Work zugeordnet, da der Ressourcenbedarf dieser Klasse am größten ist. Den Klassen Monitor und Command wurden Anteile entsprechend ihrer Bedeutung und aufgrund der gesammelten Erfahrungswerte zugeordnet. Der Klasse Report haben Sie keine Anteile zugeordnet, weil sie die einzige Unterklasse in Schicht 1 ist, so dass Zuordnungen von Anteilen irrelevant sind. Jobs in der Klasse Report können nur die Ressourcen nutzen, die von allen anderen Unterklassen in Schicht 0 nicht benutzt werden.

Im nächsten Schritt dieses Beispiels ordnen Sie Klassen ohne Anteilen Grenzwerte zu. (Sie können auch Klassen mit Anteilen Grenzwerte zuordnen. Weitere Informationen hierzu finden Sie im Abschnitt "Ressourcen mit WLM verwalten".)

5. Ordnen Sie den Superklassen Grenzwerte zu. Dazu müssen Sie die Datei `/etc/wlm/MyConfig/limits` editieren, so dass sie Folgendes enthält:

Default:

```
CPU = 0%-10%;100%  
memory = 0%-10%;100%
```

SysTools:

CPU = 0%-10%;100%
memory = 0%-5%;100%

System:

CPU = 5%-50%;100%
memory = 5%-50%;100%

Sie haben den Klassen System, SysTools und Default gemäßigte Obergrenzen zugeordnet, damit diese Klassen nicht zu stark in die übrigen Arbeitsprozesse auf dem System eingreifen können. Der Klasse System haben Sie für CPU und Speicher Untergrenzen zugeordnet, weil sie Prozesse enthält, die für den Systembetrieb unabdingbar sind. Dieser Klasse muss immer ein gewisses Maß an Ressourcen zur Verfügung stehen.

6. Ordnen Sie den Unterklassen Grenzwerte zu. Dazu müssen Sie die Dateien `/etc/wlm/MyConfig/DeptA/limits` und `/etc/wlm/MyConfig/DeptB/limits` editieren, so dass sie Folgendes enthalten:

Listen:

CPU = 10%-30%;100%
memory = 10%-20%;100%

Monitor:

CPU = 0%-30%;100%
memory = 0%-30%;100%

Anmerkung: Die Grenzwerte können für jede Unterklassendatei anders sein.

Sie haben den Klassen Listen und Monitor gemäßigte Obergrenzen zugeordnet, damit sie die übrigen Klassen derselben Superklasse nicht zu stark beeinträchtigen. Insbesondere soll vermieden werden, dass das System fortfährt, Anfragen nach Jobs in der Klasse Work entgegenzunehmen, wenn die Klasse Work keinen Zugriff auf die für die Verarbeitung erforderlichen Ressourcen hat. Der Klasse Listen wurde außerdem eine Untergrenze zugeordnet, um eine kurze Antwortzeit zu gewährleisten. Die Untergrenze für den Speicher stellt sicher, dass Seiten, die von dieser Klasse verwendet werden, nicht durch Seitenersetzung "gestohlen" werden. Dies wiederum führt zu schnelleren Ausführungszeiten. Die Untergrenze für die CPU stellt sicher, dass diese Prozesse, sofern sie ausgeführt werden können, die höchste Priorität für den Zugriff (in der Superklasse) auf die CPU-Ressourcen haben.

Schritt 3. Konfiguration von Workload Manager optimieren

1. Überwachen Sie das System mit dem Befehl `wlmstat`, und stellen Sie sicher, dass die von WLM vorgenommene Regulierung Ihren Zielen entspricht und nicht einige Anwendungen übermäßig vieler Ressourcen "beraubt", während andere Anwendungen mehr Ressourcen erhalten, als sie eigentlich sollten. In einem solchen Fall müssen Sie die Anteile anpassen und WLM aktualisieren.
2. Während Sie die Zahlen für die Anteile, Grenzwerte und Schichten überwachen, entscheiden Sie, ob Sie die Verwaltung der Unterklassen für einige oder alle Superklassen delegieren möchten. Der Administrator kann dann die Zahlen für die Anteile, Grenzwerte und Schichten überwachen und konfigurieren.

Der Administrator jeder Superklasse kann diesen Prozess für die Unterklassen jeder Superklasse wiederholen. Der einzige Unterschied besteht darin, dass WLM auf Ebene der Unterklassen nicht im passiven Modus ausgeführt werden kann. Die Konfiguration und Optimierung von Unterklassen muss vorgenommen werden, wenn WLM im aktiven Modus arbeitet. Eine Möglichkeit, die Benutzer und Anwendungen in der Klasse nicht zu beeinträchtigen, besteht darin, die jeweilige Schicht zu starten und die Anteile und Grenzwerte für die Unterklassen mit den Standardwerten ('-' (Minuszeichen) für Anteile, 0 % für Mindestwert und 100 % für veränderlichen und festen Maximalwert) zu aktivieren. Mit diesen Einstellungen findet keine Regulierung der Ressourcenzuordnung für die Unterklassen durch WLM statt.

Weitere Informationsquellen

- Workload Manager
- Workload Management

- Der Abschnitt "Workload Management Diagnosis" in der Veröffentlichung *Performance Management*
- Die Beschreibungen der Dateien `classes`, `limits`, `rules` und `shares` in der Veröffentlichung *Files Reference*
- Die Beschreibungen der Befehle `topas`, `wlmassign`, `wlmcheck`, `wlmcntrl` und `wlmstat`
- Die Beschreibungen der WLM-Subroutinen, insbesondere der Subroutine `wlm_set_tag`

Zugehörige Konzepte:

„Workload Manager konfigurieren“ auf Seite 553

Klassendefinitionen, Klassenattribute, Anteile und Grenzwerte sowie die Regeln für die automatische Klassenzuordnung können mit SMIT oder der WLM-Befehlszeilenschnittstelle eingegeben werden. Diese Definitionen und Regeln werden in unverschlüsselten Textdateien gespeichert, die auch mit einem Texteditor erstellt oder geändert werden können.

Klassen

Workload Manager unterstützt Sie bei der Steuerung der Zuordnung von Systemressourcen, indem Serviceklassen definiert und jeder dieser Klassen Ressourcen zugeordnet werden.

Jede Klasse hat eine Gruppe von Attributen, die ihre Ressourcennutzungsrechte und anderes Verhalten bestimmen. Jeder Prozess des Systems wird einer Serviceklasse zugeordnet und unterliegt damit den festgelegten Ressourcennutzungsrechten und Verhalten für diese Klasse. Prozesse können einer Klasse manuell durch manuelle Zuordnung oder automatisch auf der Basis benutzerdefinierter Klassifizierungsregeln zugeordnet werden.

WLM unterstützt zwei Ebenen von Klassen: *Superklassen* und *Unterklassen*. Superklassen werden Ressourcennutzungsrechte basierend auf den verfügbaren Systemressourcen zugeordnet. Die Nutzungsrechte für Unterklassen werden relativ zu den Nutzungsrechten der zugehörigen Superklasse festgelegt. Unterklassen können optional definiert werden, um eine differenzierte Steuerung der Prozesse in einer Superklasse zu ermöglichen. Außerdem kann die Zuständigkeit für die Definition von Unterklassen an einen Administrator oder eine Administratorgruppe für eine Superklasse delegiert werden.

Sowohl auf Superklassen- als auch auf Unterklassebene können Sie Klassen, Ressourcenanteile und -grenzwerte sowie Regeln mit SMIT oder mit der Befehlszeilenschnittstelle definieren. Anwendungen können die WLM-APIs verwenden. Konfigurationsdefinitionen werden in einer Gruppe von Textdateien, den so genannten *WLM-Eigenschaftendateien* gespeichert.

Ein Klassenname kann bis zu 16 Zeichen lang sein und darf nur Großbuchstaben, Kleinbuchstaben, Zahlen und Unterstreichungszeichen (`_`) enthalten. Die Namen der Superklassen müssen in jeder WLM-Konfiguration eindeutig sein. Jeder Unterklassenname muss innerhalb der zugehörigen Superklasse eindeutig sein, kann aber identisch mit Unterklassennamen in anderen Superklassen sein. Zur eindeutigen Identifizierung jeder Unterklasse setzt sich der vollständige Name einer Unterklasse aus dem Namen der Superklasse und dem Namen der Unterklasse zusammen, die durch einen Punkt voneinander getrennt sind, z. B. *Super.Sub*.

Superklassen:

Der Systemadministrator kann bis zu 64 Superklassen definieren.

Zusätzlich werden die folgenden fünf Superklassen automatisch erstellt:

Superklasse Default

Dies ist die Standardsuperklasse und wird immer definiert. Der Superklasse Default werden alle Prozesse zugeordnet, die keine Rootprozesse sind und nicht automatisch einer bestimmten Superklasse zugeordnet werden. Mit bestimmten Zuordnungsregeln können auch andere Prozesse der Superklasse *Default* zugeordnet werden.

Superklasse System

Dieser Superklasse werden alle berechtigten (Root-)Prozesse zugeordnet, wenn diese Prozesse

nicht durch Regeln einer bestimmten Klasse zugeordnet werden. Sie enthält außerdem die Speicherseiten, die zu Kernelspeichersegmenten und Kernelprozessen gehören. Mit bestimmten Zuordnungsregeln für diese Superklasse können auch andere Prozesse der Superklasse "System" zugeordnet werden. Diese Superklasse hat standardmäßig einen Mindestspeichergrenzwert von 1 %.

Superklasse *Shared*

Diese Superklasse erhält die Speicherseiten, die von Prozessen in mehreren Superklassen gemeinsam genutzt werden. Dazu gehören Seiten in gemeinsam genutzten Speicherbereichen und Seiten in Dateien, die von Prozessen in mehreren Superklassen (oder in Unterklassen unterschiedlicher Superklassen) verwendet werden. Gemeinsam genutzter Speicher und Dateien, die von mehreren Prozessen verwendet werden, die alle zu einer einzelnen Superklasse (oder Unterklassen derselben Superklasse) gehören, werden dieser Superklasse zugeordnet. Die Seiten werden erst dann der Superklasse *Shared* zugeordnet, wenn ein Prozess aus einer anderen Superklasse auf den gemeinsam genutzten Speicherbereich oder die gemeinsam genutzte Datei zugreift. Für diese Superklasse können nur Anteile und Grenzwerte für den physischen Speicher definiert werden. Es ist nicht möglich, Anteile oder Grenzwerte für die anderen Ressourcentypen, Unterklassen oder Zuordnungsregeln anzugeben. Ob ein Speichersegment, das von Prozessen in unterschiedlichen Unterklassen derselben Superklasse der Unterklasse *Shared* zugeordnet wird oder in seiner ursprünglichen Unterklasse verbleibt, richtet sich nach dem Wert des Attributs **localshm** der ursprünglichen Unterklasse.

Superklasse *Unclassified*

Diese Superklasse ist eine Hauptspeicherzuordnung für nicht klassifizierte Prozesse. Die Prozesse, die beim Starten von WLM vorhanden sind, werden entsprechend den Zuordnungsregeln der geladenen WLM-Konfiguration klassifiziert. Während dieser ersten Klassifizierung werden alle Speicherseiten, die zu den einzelnen Prozessen gehören, entweder der Superklasse, zu der der Prozess gehört (bei nicht gemeinsamer Nutzung oder gemeinsamer Nutzung durch Prozesse derselben Superklasse) oder der Superklasse *Shared* (bei gemeinsamer Nutzung durch Prozesse in unterschiedlichen Superklassen) zugeordnet.

Einige Seiten können zu diesem Zeitpunkt der Klassifizierung jedoch nicht direkt an Prozesse (und damit an eine Klasse) gebunden werden. In diesem Fall wird der Speicher der Superklasse *Unclassified* zugeordnet. Der Großteil dieses Hauptspeichers wird mit der Zeit erneut korrekt klassifiziert, wenn ein Prozess darauf zugreift oder wenn der Hauptspeicher nach dem Starten von WLM freigegeben und erneut zugeordnet wird. Es gibt keine Prozesse in der Superklasse *Unclassified*. Für diese Superklasse können Anteile und Grenzwerte für den physischen Speicher definiert werden. Es ist nicht möglich, Anteile oder Grenzwerte für die anderen Ressourcentypen, Unterklassen oder Zuordnungsregeln anzugeben.

Superklasse *Unmanaged*

Eine spezielle Superklasse mit dem Namen *Unmanaged* wird immer definiert. Dieser Klasse werden keine Prozesse zugeordnet. Diese Klasse erfasst die Speicherbelegung für alle festgehaltenen Seiten im System, die nicht von WLM verwaltet werden. Die CPU-Auslastung für *waitproc*-Prozesse wird in keiner Klasse erfasst, um zu vermeiden, dass die Systemauslastung bei 100 % zu sein scheint. Es ist nicht möglich, für diese Superklasse Anteile oder Grenzwerte für die anderen Ressourcentypen, Unterklassen oder Zuordnungsregeln anzugeben.

Unterklassen:

Der Systemadministrator oder ein Superklassenadministrator kann bis zu 61 Unterklassen definieren.

Zusätzlich sind immer zwei spezielle Unterklassen, *Default* und *Shared*, definiert.

Unterklasse *Default*

Dies ist die Standardunterklasse und ist immer definiert. Der Unterklasse *Default* werden alle Prozesse zugeordnet, die nicht automatisch einer bestimmten Unterklasse der Superklasse zugeordnet werden. Sie können der Unterklasse *Default* auch andere Prozesse zuordnen, indem Sie bestimmte Zuordnungsregeln definieren.

Unterklasse *Shared*

Diese Unterklasse erhält alle Speicherseiten, die von Prozessen in mehreren Unterklassen der Superklasse verwendet werden. Dazu gehören Seiten in gemeinsam genutzten Speicherbereichen und Seiten in Dateien, die von Prozessen in mehreren Unterklassen derselben Superklasse verwendet werden. Gemeinsam genutzter Speicher und Dateien, die von mehreren Prozessen verwendet werden, die alle zu derselben Unterklasse gehören, werden dieser Unterklasse zugeordnet. Die Seiten werden erst dann der Unterklasse *Shared* der Superklasse zugeordnet, wenn ein Prozess aus einer anderen Unterklasse derselben Superklasse auf den gemeinsam genutzten Speicherbereich oder die gemeinsam genutzte Datei zugreift. Die Unterklasse *Shared* enthält keine Prozesse. Für diese Unterklasse können nur Anteile und Grenzwerte für den physischen Hauptspeicher angegeben werden. Es ist nicht möglich, Anteile oder Grenzwerte für andere Ressourcentypen oder Zuordnungsregeln für diese Unterklasse anzugeben. Ob ein Speichersegment, das von Prozessen in unterschiedlichen Unterklassen derselben Superklasse der Unterklasse *Shared* zugeordnet wird oder in seiner ursprünglichen Unterklasse verbleibt, richtet sich nach dem Wert des Attributs **localshm** der ursprünglichen Unterklasse.

Klassenattribute:

Sie können alle Attribute einer WLM-Klasse auflisten.

Klassenname

Der Klassenname kann bis zu 16 Zeichen lang sein und darf nur Großbuchstaben, Kleinbuchstaben, Zahlen und Unterstreichungszeichen () enthalten.

Schicht

Eine Zahl zwischen 0 und 9, die verwendet wird, um Prioritäten für die Zuordnung von Ressourcen zu Klassen zu vergeben.

Vererbung

Gibt an, ob ein untergeordneter Prozess die Klassenzuordnung von seinem übergeordneten Prozess erbt.

localshm

Verhindert, dass Speichersegmente, die zu einer Klasse gehören, der Klasse *Shared* zugeordnet werden.

Administrator (*adminuser*, *admingroup*, *authgroup*) (nur Superklassen)

Delegiert die Verwaltung einer Superklasse.

Berechtigung (*authuser*, *authgroup*)

Delegiert das Recht für die manuelle Zuordnung eines Prozesses zu einer Klasse.

Ressourcengruppe (*rset*)

Beschränkt die Ressourcen, auf die eine bestimmte Klasse Zugriff hat, auf der Basis von CPUs (Prozessorgruppe) ein.

delshm

Löscht die gemeinsam genutzten Speichersegmente, wenn der letzte referenzierende Prozess aufgrund der Begrenzung des virtuellen Speichers beendet wird.

vmeforce

Gibt an, ob alle Prozesse in einer Klasse oder nur der Prozess, der gegen den Grenzwert verstößt, beendet wird, wenn eine Klasse ihren Grenzwert für den virtuellen Speicher erreicht.

io_priority

Gibt die Priorität an, die E/A-Anforderungen zugeordnet wird, die von Threads ausgegeben werden, die dieser Klasse zugeordnet sind. Diese Priorität wird verwendet, um Prioritäten für E/A-Puffer auf Einheitenebene zu vergeben. Wenn die Speichereinheit keine E/A-Prioritäten unterstützt, wird die Priorität ignoriert. Die gültigen E/A-Prioritäten sind 0 bis 15.

Zugehörige Konzepte:

„Zuordnung von Prozessen zu Klassen für das Workload-Management“ auf Seite 517

Die Prozesse werden unter Verwendung der vom Systemadministrator definierten Klassenzuordnungsregeln einer Klasse zugeordnet. Die Klassifizierungskriterien basieren auf dem Wert einer Gruppe von Prozessattributen, wie z. B. der Benutzer-ID, der Gruppen-ID, dem Namen der Anwendungsdatei, dem Prozessstyp oder der Prozesskennung.

Schichtenattribut:

Schichten stellen die Reihenfolge dar, in der Systemressourcen zu WLM-Klassen zugeordnet werden.

Der Administrator kann Klassen in bis zu 10 Schichten, nummeriert von 0 bis 9, definieren, wobei 0 die höchste bzw. wichtigste Schicht ist. Die der Schicht 0 zur Verfügung stehende Ressourcenmenge entspricht der Summe aller verfügbaren Systemressourcen. Die den niedrigeren Schichten (mit den höheren Nummern) zur Verfügung stehende Ressourcenmenge entspricht der Summe der Ressourcen, die von allen höheren Schichten nicht verwendet werden. Die Zielprozentsätze für den Verbrauch für Klassen basieren auf der Anzahl der aktiven Anteile in der jeweiligen Schicht und der Ressourcenmenge, die der Schicht zur Verfügung steht. Da die Schicht 0 die einzige Schicht ist, der jederzeit garantiert Ressourcen zur Verfügung stehen, wird empfohlen, dass die Prozesse, die für den Systembetrieb entscheidend sind, einer Klasse in dieser Schicht zugeordnet werden. Wenn für eine Klasse kein Schichtenwert angegeben wird, wird die Klasse der Schicht 0 zugeordnet.

Eine Schicht kann auf Superklassen- und Unterklassenebene angegeben werden. Superklassenschichten werden verwendet, um die Priorität für die Ressourcenzuordnung zwischen Superklassen zu regeln. Unterklassenschichten werden verwendet, um die Priorität für die Ressourcenordnung zwischen Unterklassen einer Superklasse zu regeln. Es gibt keine Beziehung zwischen den Unterschichten unterschiedlicher Superklassen.

Attribut inheritance:

Das Attribut **inheritance** einer Klasse gibt an, ob Prozesse in der Klasse automatisch erneut klassifiziert werden sollen, wenn sich eines der Klassifizierungsattribute des Prozesses ändert.

Wenn ein neuer Prozess mit der Subroutine **fork** erstellt wird, erbt er automatisch die Klasse des übergeordneten Prozesses, egal ob die Vererbung aktiviert ist oder nicht. Ausgenommen hiervon ist der Fall, dass der übergeordnete Prozesse ein Attribut besitzt, dass sein Attribut "Attribut bei Verzweigung erben" (**inherit tag at fork**) inaktiviert ist oder dass die Klassenvererbung für die Klasse des übergeordneten Prozesses inaktiviert ist. In diesem Fall wird der untergeordnete Prozess entsprechend den Klassifizierungsregeln erneut klassifiziert.

Wenn die Vererbung für eine Klasse nicht aktiviert ist, wird jeder Prozess in der Klasse automatisch entsprechend den Klassifizierungsregeln klassifiziert, nachdem ein Service aufgerufen wurde, der ein in der Regel verwendetes Prozessattribut ändert. Zu den häufigsten Vertretern dieser Aufrufe gehört die Subroutine **exec**. Es gibt aber auch weitere Subroutinen, die die Klassifizierung ändern können, z. B. **setuid**, **setgid**, **plock**, **setpri** und **wlm_set_tag**. Wenn die Vererbung aktiviert ist, wird der Prozess nicht basierend auf den Klassifizierungsregeln erneut klassifiziert und verbleibt in seiner aktuellen Klasse. Die manuelle Zuordnung hat Priorität vor der Vererbung und kann verwendet werden, um Prozesse erneut zu klassifizieren, die sich in einer Klasse befinden, in der die Vererbung aktiviert ist.

Die gültigen Werte für das Attribut **inheritance** sind **yes** und **no**. Wenn dieses Attribut nicht angegeben wird, wird die Vererbung für eine Klasse nicht aktiviert.

Dieses Attribut kann auf Superklassen- und Unterklassenebene angegeben werden. Für eine Unterklasse einer bestimmten Superklasse gilt Folgendes:

- Wenn das Attribut **inheritance** auf Superklassen- und Unterklassenebene den Wert **yes** hat, verbleibt ein untergeordneter Prozess eines Prozesses in derselben Unterklasse.

- Wenn das Attribut **inheritance** auf Superklassenebene den Wert **yes** und für die Unterklasse den Wert **no** hat oder nicht angegeben ist, verbleibt ein untergeordneter Prozess eines Prozesses in der Unterklasse in derselben Superklasse und wird entsprechend den Zuordnungsregeln für die Superklasse einer der zugehörigen Unterklassen zugeordnet.
- Wenn das Attribut **inheritance** auf Superklassenebene den Wert **no** hat oder nicht angegeben ist und auf Unterklassenebene den Wert **yes** hat, wird ein untergeordneter Prozess eines Prozesses in der Unterklasse an die automatischen Zuordnungsregeln für die Superklassen übergeben.
 - Wenn der Prozess von den Regeln in derselben Superklasse klassifiziert wird, verbleibt er in der Unterklasse (d. h., er wird nicht an die Zuordnungsregeln der Unterklasse übergeben).
 - Wenn der Prozess von den Regeln der Superklasse einer anderen Superklasse zugeordnet wird, werden die Unterklassenzuordnungsregeln der neuen Superklasse angewendet, um die Unterklasse der neuen Superklasse zu bestimmen, der der Prozess zugeordnet wird.
- Wenn das Attribut **inheritance** für Superklasse und Unterklasse jeweils den Wert **no** hat oder nicht angegeben ist, wird ein untergeordneter Prozess eines Prozesses in der Unterklasse an die automatische Standardzuordnung übergeben.

Attribut localshm:

Das Attribut **localshm** kann auf Superklassen- und Unterklassenebene angegeben werden.

Mit dem Attribut **localshm** kann verhindert werden, dass Speichersegmente, die zu einer Klasse gehören, beim Zugriff durch Prozesse in anderen Klassen der Superklasse oder Unterklasse *Shared* zugeordnet werden. Die gültigen Werte für das Attribut sind **yes** und **no**. Der Wert **yes** bedeutet, dass gemeinsam genutzte Speichersegmente in dieser Klasse lokale Segmente der Klasse bleiben müssen und nicht der entsprechenden Klasse *Shared* zugeordnet werden. Wenn das Attribut nicht angegeben ist, wird als Standardwert **no** angenommen.

Speichersegmente werden nach Fehlseitenbedingungen klassifiziert. Beim Erstellen eines Segments wird es als zur Superklasse *Unclassified* gehörig gekennzeichnet. Bei der ersten Fehlseitenbedingung für das Segment wird dieses Segment derselben Klasse zugeordnet, zu der auch der fehlerhafte Prozess gehört. Sollte später in einem Prozess, der zu einer anderen Klasse als das Segment gehört, eine Fehlseitenbedingung für dieses Segment auftreten, prüft WLM, ob das Segment erneut klassifiziert und der Klasse *Shared* (Superklasse oder Unterklasse) zugeordnet werden muss. Gehören der fehlerhafte Prozess und das Segment zu unterschiedlichen Superklassen, tritt einer der folgenden Fälle ein:

- Wenn das Attribut **localshm** der Segmentsuperklasse auf **yes** gesetzt ist, verbleibt das Segment in seiner aktuellen Superklasse. Wenn das Attribut **localshm** der Segmentunterklasse auf **yes** gesetzt ist, verbleibt das Segment in seiner aktuellen Unterklasse. Wenn das Attribut **localshm** der Superklasse auf **yes**, aber das Attribut der Unterklasse auf **no** gesetzt ist, wird das Segment der Unterklasse *Shared* der aktuellen Superklasse zugeordnet.
- Wenn das Attribut **localshm** der Segmentsuperklasse auf **no** gesetzt ist, wird das Segment der Superklasse *Shared* zugeordnet. Dies ist die Standardaktion.

Wenn der fehlerhafte Prozess und das Segment zu unterschiedlichen Unterklassen derselben Superklasse gehören und das Attribut **localshm** der Segmentunterklasse auf **yes** gesetzt ist, verbleibt das Segment in der aktuellen Klasse (Superklasse und Unterklasse). Andernfalls wird das Segment der Unterklasse *Shared* der Superklasse zugeordnet.

Das Segment wird unabhängig vom Wert des Attributs **localshm** natürlich nicht erneut klassifiziert, wenn der fehlerhafte Prozess und das Segment zu derselben Klasse gehören (derselben Superklasse und derselben Unterklasse).

Administratorattribut:

Die Attribute **adminuser** und **admingroup** werden verwendet, um die Verwaltung der Superklasse an einen Benutzer oder eine Benutzergruppe zu delegieren.

Anmerkung: Diese Attribute sind nur für Superklassen gültig.

Das Attribut **adminuser** gibt den Namen des Benutzers (wie in `/etc/passwd` aufgelistet) an, der berechtigt ist, Verwaltungstasks für die Superklasse auszuführen. Das Attribut **admingroup** gibt den Namen der Benutzergruppe (wie in `/etc/group` aufgelistet) an, die berechtigt ist, Verwaltungstasks für die Superklasse auszuführen.

Für jedes Attribut kann nur ein Wert (Benutzer oder Gruppe) angegeben werden. Sie können eines der beiden Attribute, keines oder beide Attribute angeben. Der Benutzer bzw. die Gruppe ist für folgende Tasks berechtigt:

- Unterklassen erstellen und löschen
- Attribute, Ressourcenanteile und Ressourcengrenzwerte für die Unterklassen ändern
- Zuordnungsregeln für die Unterklasse definieren, entfernen oder ändern
- Aktive WLM-Konfiguration für die Superklasse aktualisieren

Berechtigungsattribut:

Die Attribute **authuser** und **authgroup** sind für alle Klassen gültig. Mit diesen Attributen wird der Benutzer bzw. die Gruppe angegeben, der bzw. die berechtigt ist, der Klasse (Superklasse oder Unterklasse) Prozesse manuell zuzuordnen.

Wenn ein Prozess (oder eine Gruppe von Prozessen) manuell einer Superklasse zugeordnet wird, werden die Zuordnungsregeln für die Superklasse verwendet, um festzustellen, zu welcher Unterklasse der Superklasse jeder einzelne Prozess zugeordnet wird.

Für jedes Attribut kann nur ein Wert (Benutzer oder Gruppe) angegeben werden. Sie können eines der beiden Attribute, keines oder beide Attribute angeben.

Ressourcengruppenattribut:

Das Ressourcengruppenattribut (genannt *rset*) kann für jede Klasse angegeben werden. Der Wert dieses Attributs ist der Name einer Ressourcengruppe, die vom Systemadministrator definiert wird.

Das Attribut *rset* stellt eine Untergruppe der auf dem System verfügbaren CPU-Ressourcen (Prozessorgruppe) dar. Die Standardeinstellung ist "system" und lässt den Zugriff auf alle verfügbaren CPU-Ressourcen des Systems zu. Es ist nur einige Einschränkung zu beachten: Wenn ein *rset* für eine Unterklasse angegeben ist, muss die CPU-Gruppe in der Gruppe eine Untergruppe der CPUs sein, die der Superklasse zur Verfügung stehen. (Ausführliche Informationen finden Sie in der Beschreibung des Befehls **mkrset** in der Veröffentlichung "Commands Reference".)

Anmerkung: Gehen Sie besonders sorgfältig bei der Zuordnung von Ressourcengruppen zu Klassen vor, die sich nicht in Schicht 0 befinden. Da untere Schichten nur Zugriff auf die Ressourcen haben, die nicht von den höheren Schichten verwendet werden, könnte eine Beschränkung einer nicht zur Schicht 0 gehörenden Klasse auf eine Untergruppe der CPUs im System zu einer Blockierung führen, wenn keine CPU-Zeit in diesen CPUs verfügbar ist.

Prozessklassifizierungen in Workload Manager

In WLM können Prozesse auf zwei Arten klassifiziert werden.

- Ein Prozess wird automatisch auf der Basis der Zuordnungsregeln zugeordnet, wenn sich die Attribute für die Prozessklassifizierung ändern. Wenn WLM im aktiven Modus ausgeführt wird, ist diese automatische Zuordnung immer aktiv (und kann nicht inaktiviert werden). Dies ist die am häufigsten angewendete Methode für die Klassifizierung von Prozessen.
- Ein ausgewählter Prozess oder eine ausgewählte Gruppe von Prozessen kann einer Klasse von einem Benutzer, der die erforderlichen Berechtigungen für die Prozesse und die Zielklasse besitzt, manuell zugeordnet werden. Für die manuelle Zuordnung kann ein WLM-Befehl, der direkt oder über SMIT auf-

gerufen werden kann, oder eine Anwendung verwendet werden, die eine Funktion der Anwendungsprogrammierschnittstelle von WLM verwendet. Diese manuelle Zuordnung überschreibt die automatische Zuordnung und Vererbung.

Automatische Klassenzuordnung in Workload Manager:

Für die automatische Zuordnung von Prozessen zu Klassen wird eine Gruppe von Klassenzuordnungsregeln verwendet, die von einem WLM-Administrator definiert werden.

Es gibt zwei Ebenen von Zuordnungsregeln:

- Eine Gruppe von Zuordnungsregeln auf WLM-Konfigurationsebene, mit denen bestimmt wird, welcher Superklasse ein bestimmter Prozess zugeordnet wird.
- Jede Superklasse mit definierten Unterklassen wiederum hat eine Gruppe von Regeln, mit denen bestimmt wird, welcher Unterklasse der Superklasse der Prozess zugeordnet wird.

Die Zuordnungsregeln auf beiden Ebenen basieren auf den Werten einer Gruppe von Prozessattributen. Diese Attribute sind im Folgenden aufgelistet:

- Prozessbenutzer-ID
- Prozessgruppen-ID
- Pfadname der ausgeführten Anwendung (Programm)
- Typ des Prozesses (z. B. 32-Bit- oder 64-Bit-Prozess)
- Prozesskennung

Die Kennung ist ein Prozessattribut, das als Zeichenfolge definiert wird und das eine Anwendung über ein Programm mit der WLM-API setzen kann.

Die Klassifizierung wird durchgeführt, sobald sich ein Attribut ändert. Hierfür werden die Werte dieser Prozessattribute mit Listen gültiger Werte verglichen, die in der Datei mit den Klassenzuordnungsregeln (*rules*) angegeben sind. Bei dem Vergleich wird festgestellt, welche Regel auf die aktuellen Werte der Prozessattribute zutrifft.

Eine Klassenzuordnungsregel ist eine Textzeichenfolge, die die folgenden Felder, jeweils durch mindestens ein Leerzeichen voneinander getrennt, enthält:

| Eintrag | Beschreibung |
|------------|---|
| Name | Dieses Feld muss den Namen einer Klasse enthalten, die in der Klassendatei definiert ist, die der Ebene der Datei <i>rules</i> entspricht (Superklasse oder Unterklasse). Klassennamen dürfen nur Groß- und Kleinbuchstaben, Zahlen und Unterstreichungszeichen enthalten und maximal 16 Zeichen lang sein. Für die systemdefinierten Klassen <i>Unclassified</i> , <i>Unmanaged</i> und <i>Shared</i> kann keine Zuordnungsregel angegeben werden. |
| Reserviert | Dieses Feld ist für künftige Erweiterungen reserviert. Dieses Feld muss ein Minuszeichen (-) enthalten. |
| Benutzer | Dieses Feld kann ein Minuszeichen (-) oder mindestens einen gültigen Benutzernamen (der in der Datei <i>/etc/passwd</i> definiert ist) enthalten. Die Liste setzt sich aus einem oder mehreren Namen zusammen, die jeweils durch ein Komma (,) voneinander getrennt sind. Einem Namen kann ein Ausrufezeichen (!) vorangestellt werden, wenn der zugehörige Benutzer aus der Klasse ausgeschlossen werden soll. Es können auch Muster in der vollständigen Korn-Shellsyntax für Mustererkennung angegeben werden, um eine Gruppe von Benutzernamen zu erfassen. Wenn keine gültigen Benutzernamen vorhanden sind, wird die Regel ignoriert. |
| Gruppe | Dieses Feld kann ein Minuszeichen (-) oder mindestens einen gültigen Gruppennamen (der in der Datei <i>/etc/group</i> definiert ist) enthalten. Die Liste setzt sich aus einem oder mehreren Namen zusammen, die jeweils durch ein Komma (,) voneinander getrennt sind. Einem Namen kann ein Ausrufezeichen (!) vorangestellt werden, wenn die zugehörige Gruppe aus der Klasse ausgeschlossen werden soll. Es können auch Muster in der vollständigen Korn-Shellsyntax für Mustererkennung angegeben werden, um eine Gruppe von Gruppennamen zu erfassen. Wenn keine gültigen Gruppennamen vorhanden sind, wird die Regel ignoriert. |

| Eintrag | Beschreibung |
|----------------|--|
| Anwendung | <p>Dieses Feld kann ein Minuszeichen (-) oder eine Liste mit den Pfadnamen von Anwendungen enthalten. Es gibt die Pfadnamen der Anwendungen (Programme) an, die von den in der Klasse enthaltenen Prozessen ausgeführt werden. Für die Anwendungsnamen können vollständige Pfadnamen oder Korn-Shellmuster angegeben werden, die eine Gruppe von Pfadnamen erfassen. Die Liste setzt sich aus einem oder mehreren Pfadnamen zusammen, die jeweils durch ein Komma (,) voneinander getrennt sind. Einem Namen kann ein Ausrufezeichen (!) vorangestellt werden, wenn eine bestimmte Anwendung ausgeschlossen werden soll.</p> <p>Zur Ladezeit muss mindestens eine Anwendung aus der Liste gefunden werden, ansonsten wird die Regel ignoriert. Regeln, die aus diesem Grund zunächst ignoriert werden, können zu einem späteren Zeitpunkt wirksam werden, wenn ein Dateisystem angehängt wird, das eine oder mehrere Anwendungen aus der Liste enthält.</p> |
| Typ | <p>Dieses Feld kann ein Minuszeichen (-) oder eine Liste mit Prozessattributen enthalten. Die gültigen Werte für diese Attribute sind im Folgenden aufgelistet:</p> <ul style="list-style-type: none"> • 32bit: Der Prozess ist ein 32-Bit-Prozess. • 64bit: Der Prozess ist ein 64-Bit-Prozess. • plock: Der Prozess hat die Subroutine plock zum Festhalten von Hauptspeicher aufgerufen. • fixed: Der Prozess ist ein Prozess mit fester Priorität (SCHED_FIFO oder SCHED_RR). <p>Der Typ fixed ist nur für Klassifizierungszwecke bestimmt. Die Prozessornutzung von Prozessen oder Threads mit fester Priorität wird von WLM nicht reguliert. Prozesse mit fester Priorität können anderen Prozessen in einer Klasse Ressourcen entziehen. Deshalb wird dieses Klassifizierungsattribut bereitgestellt, um solche Jobs isolieren zu können. Dieses Attribut kann auch verwendet werden, um den Verbrauch solcher Prozesse zurückzumelden.</p> <p>Der Wert des Feldes Typ kann eine Kombination aus einem oder mehreren der zuvor beschriebenen Attribute sein. Die einzelnen Attribute müssen durch ein Pluszeichen (+) voneinander getrennt werden. Die Werte 32bit und 64bit schließen sich gegenseitig aus.</p> |
| Kennung | <p>Dieses Feld kann ein Minuszeichen (-) oder eine Liste mit Anwendungskennungen enthalten. Eine Anwendungskennung ist eine Zeichenfolge mit bis zu 30 alphanumerischen Zeichen. Die Liste setzt sich aus einem oder mehreren Anwendungskennungen zusammen, die jeweils durch ein Komma voneinander getrennt sind.</p> |

Die Attribute Benutzer, Gruppe, Anwendung und Kennung können eine Attributwertgruppierung sein.

Wenn ein Prozess (mit fork) erstellt wird, bleibt er in derselben Klasse wie der übergeordnete Prozess. Es findet eine erneute Klassifizierung statt, wenn der neue Prozess einen Systemaufruf absetzt, der eines der Attribute des Prozesses ändern könnte, die für die Klassifizierung verwendet werden. Beispiele für solche Systemaufrufe sind *exec*, *setuid* (und zugehörige Aufrufe), *setgid* (und zugehörige Aufrufe), *setpri* und *plock*.

Für die Klassifizierung des Prozesses untersucht WLM die Datei *rules* der Ausgangsebene für die aktive Konfiguration, um festzustellen, zu welcher Superklasse der Prozess gehört. WLM vergleicht für jede Regel in der Datei die aktuellen Werte der Prozessattribute mit den Werten und Listen, die in der Regel angegeben sind. Regeln werden in der Reihenfolge geprüft, in der sie in der Datei angegeben sind. Bei einer Übereinstimmung wird der Prozess der Superklasse zugeordnet, die im ersten Feld der Regel angegeben ist. Anschließend wird die Datei *rules* für die Superklasse auf dieselbe Weise untersucht, um festzustellen, welcher Unterklasse der Prozess zugeordnet werden muss.

Ein Prozess stimmt mit einer der Regeln überein, wenn jedes seiner Attribute mit dem entsprechenden Feld in der Regel übereinstimmt. In der folgenden Liste sind die Kriterien aufgeführt, mit denen festgestellt wird, ob der Wert eines Attributs mit dem Wert im entsprechenden Feld in der Datei *rules* übereinstimmt:

- Wenn das Feld in der Datei *rules* ein Minuszeichen (-) enthält, wird jeder Wert des entsprechenden Prozessattributs als Übereinstimmung betrachtet.
- Für alle Attribute mit Ausnahme von *Typ* gilt Folgendes: Wenn der Wert des Prozessattributs mit einem der in der Liste in der Datei *rules* aufgeführten Werte übereinstimmt, der nicht ausgeschlossen ist (dem ein Ausrufezeichen (!) vorangestellt ist), liegt eine Übereinstimmung vor.

- Für das Attribut *Typ* gilt Folgendes: Wenn sich einer der in der Regel aufgeführten Werte aus zwei oder mehr Werten zusammensetzt, die durch ein Pluszeichen (+) voneinander getrennt sind, liegt nur dann eine Übereinstimmung mit einem Prozess vor, wenn die Merkmale des Prozesses mit allen Werten übereinstimmen.

Sowohl auf Superklassen- als auch auf Unterklassenebene arbeitet WLM die Regeln in der Reihenfolge ab, in der sie in der Datei `rules` aufgeführt sind, und ordnet den Prozess der Klasse zu, die der ersten übereinstimmenden Regel entspricht. Die Reihenfolge der Regeln in der Datei `rules` ist deshalb von entscheidender Bedeutung. Gehen Sie beim Erstellen oder Ändern der Datei `rules` mit Vorsicht vor.

Manuelle Klassenzuordnung in Workload Manager:

Prozesse und Prozessgruppen können mit SMIT oder mit dem Befehl `wlmassign` manuell einer Superklasse und/oder einer Unterklasse zugeordnet werden.

Weitere Informationen finden Sie in der Beschreibung des Befehls `wlmassign` in der Veröffentlichung "Commands Reference". Eine Anwendung kann Prozesse mit der API-Funktion `wlm_assign` zuordnen.

Für die manuelle Zuordnung von Prozessen zu Klassen und das Zurücknehmen einer vorhandenen Zuordnung muss ein Benutzer die entsprechenden Berechtigungen haben. Eine manuelle Zuordnung kann gesondert auf Superklassenebene oder Unterklassenebene oder auf beiden Ebenen vorgenommen und rückgängig gemacht werden. Diese Zuordnung wird mit Flags für die Programmierschnittstelle und eine Gruppe von Optionen für die Befehlszeilenschnittstelle angegeben, die von den WLM-Verwaltungstools verwendet wird. Ein Prozess kann also manuell einer Superklasse oder einer Unterklasse ausschließlich oder einer Superklasse und einer Unterklasse dieser Superklasse zugeordnet werden. Im letzten Fall kann diese doppelte Zuordnung gleichzeitig (mit einem einzigen Befehl oder API-Aufruf) oder zu unterschiedlichen Zeiten von unterschiedlichen Benutzern vorgenommen werden.

Die Zuordnung ist sehr flexibel, kann aber auch verwirrend sein. Im Folgenden werden zwei Beispiele für die möglichen Szenarios beschrieben.

Beispiel 1: Erste Zuordnung von Prozessen

Ein Systemadministrator ordnet den *Prozess1* manuell von *SuperklasseA* zu *SuperklasseB* zu (reine Zuordnung auf Superklassenebene). Die Regeln für die automatische Zuordnung der Unterklassen von *SuperklasseB* werden von WLM verwendet, um festzustellen, welcher Unterklasse der Prozess letztendlich zugeordnet wird. *Prozess1* wird der Superklasse *SuperklasseB.UnterklasseA* zugeordnet und so gekennzeichnet, dass deutlich wird, dass er eine exklusive Zuordnung zu einer Superklasse hat.

Ein Benutzer mit den erforderlichen Berechtigungen ordnet *Prozess2* von der aktuellen Klasse *SuperklasseA.UnterklasseA* zu einer neuen Unterklasse derselben Superklasse zu, *SuperklasseA.UnterklasseB*. *Prozess2* wird der neuen Unterklasse zugeordnet und so gekennzeichnet, dass deutlich wird, dass er eine exklusive Zuordnung zu einer Unterklasse hat.

Ein WLM-Administrator für die Unterklassen der *SuperklasseB* ordnet *Prozess1* manuell der *UnterklasseC* zu, die eine weitere Unterklasse der *SuperklasseB* ist. *Prozess1* wird erneut klassifiziert und *SuperklasseB.UnterklasseC* zugeordnet und jetzt so gekennzeichnet, dass deutlich wird, dass er eine Zuordnung auf Superklassen- und eine auf Unterklassenebene hat.

Beispiel 2: Erneute Zuordnung oder Zurücknehmen einer manuellen Zuordnung

Die erneute Zuordnung und das Zurücknehmen einer manuellen Zuordnung auf Unterklassenebene ist weniger komplex und betrifft nur die Zuordnung auf Unterklassenebene.

Angenommen, der Systemadministrator möchte *Prozess2* einer Superklasse mit mehr Ressourcen zuordnen, und entscheidet, *Prozess2* der *SuperklasseC* zuzuordnen. In Beispiel 1 wurde *Prozess2* manuell durch eine exklusive Unterklassenzuordnung der *UnterklasseB* von *SuperklasseA* zugeordnet. Da *Prozess2* einer anderen Superklasse zugeordnet wird, wird die vorherige manuelle Zuordnung bedeutungslos und wird rückgängig gemacht. *Prozess2* hat jetzt eine manuelle exklusive Su-

perklassenzuordnung zur *SuperklasseC* und wird aufgrund des fehlenden Vererbungsattributs (inheritance) unter Verwendung der Regeln für automatische Zuordnung einer Unterklasse der *SuperklasseC* zugeordnet.

Jetzt beschließt der Systemadministrator, die manuelle Zuordnung von *Prozess1* zur *SuperklasseB* aufzulösen. Die manuelle Zuordnung auf Superklassenebene von *Prozess1* wird rückgängig gemacht, und aufgrund des fehlenden Vererbungsattributs (inheritance) wird *Prozess1* unter Verwendung der Regeln für automatische Zuordnung der Ausgangsebene einer Superklasse zugeordnet.

Wenn die Regeln nicht geändert wurden, wird *Prozess1* der *SuperklasseA* zugeordnet, und die manuelle Zuordnung des Prozesses auf Unterklasse zu *SuperklasseB.UnterklasseC* wird bedeutungslos und deshalb rückgängig gemacht.

Wenn *Prozess1* aus irgendeinem Grund von den Regeln der Ausgangsebene der *SuperklasseB* zugeordnet wird, bleibt die Zuordnung auf Unterklassenebene zu *SuperklasseB.UnterklasseC* gültig und wirksam. *Prozess1* hat jetzt eine manuelle exklusive Unterklassenzuordnung.

Aktualisierungen in Workload Manager:

Wenn WLM (mit dem Befehl **wlmcntrl -u**) aktualisiert wird, kann die aktualisierte Konfiguration eine neue Gruppe von Klassifizierungsregeln laden.

In einem solchen Fall werden Prozesse häufig anhand der neuen Regeln erneut klassifiziert. WLM führt keine erneute Klassifizierung von Prozessen durch, die manuell zugeordnet wurden oder die sich in einer Klasse mit aktivierter Vererbung befinden, es sei denn, ihre Klasse ist in der neuen Konfiguration nicht vorhanden.

Sicherheitsaspekte zu Workload Manager:

Wenn der Benutzer einen Prozess zu einer Klasse zuordnen oder eine frühere manuelle Zuordnung aufheben möchte, muss er Berechtigungen für den Prozess und die Zielklasse haben.

Diese Einschränkungen lassen sich in die folgenden Regeln umsetzen:

- Root kann jeden Prozess zu jeder Klasse zuordnen.
- Ein Benutzer mit Verwaltungsberechtigungen für die Unterklassen einer bestimmten Superklasse (d. h., der Benutzer- oder Gruppenname stimmt mit dem Benutzer- bzw. Gruppennamen in den Attributen **adminuser** und **admingroup** der Superklasse überein) kann jeden Prozess, der einer der Unterklassen dieser Superklasse zugeordnet ist, einer anderen Unterklasse der Superklasse zuordnen.
- Benutzer können ihre eigenen Prozesse (Prozesse, die derselben realen oder aktuellen Benutzer-ID zugeordnet sind) zu einer Unterklasse zuordnen, für die sie manuelle Zuordnungsberechtigung haben (d. h., der Benutzer- oder Gruppenname stimmt mit dem Benutzer- bzw. Gruppennamen in den Attributen **authuser** und **authgroup** der Superklasse oder Unterklasse überein).

Wenn Sie eine manuelle Zuordnung ändern oder aufheben möchten, müssen Sie mindestens den Berechtigungsstand der Person haben, die die letzte manuelle Zuordnung vorgenommen hat.

Ressourcenverwaltung mit Workload Manager

WLM überwacht und reguliert die Ressourcenauslastung durch die im System aktiven Threads und Prozesse auf Klassenbasis. Sie können Minimal- und Maximalwerte pro Klasse für jeden von WLM verwalteten Ressourcentyp sowie einen Zielwert pro Klasse für jede Ressource festlegen.

Dieser Zielwert steht stellvertretend für den Anteil der Ressource, der für die Jobs in der Klasse optimal ist. Die Anteile und Grenzwerte auf Superklassenebene beziehen sich auf die Gesamtmenge jeder im System verfügbaren Ressource. Auf Unterklassenebene beziehen sich Anteile und Grenzwerte auf die Menge jeder Ressource, die der Superklasse, zu der die Unterklasse gehört, zur Verfügung gestellt wird (Superklassenziel). Die Klassenhierarchie ist eine Möglichkeit, die Systemressourcen auf die Benutzergruppen (Superklassen) zu verteilen und die Verwaltung dieser Ressourcenanteile an die Administratoren der Su-

perklassen zu delegieren. Jeder Superklassenadministrator kann diese Ressourcenmenge auf die Benutzer in der Gruppe umverteilen, indem er Unterklassen erstellt und Ressourcenberechtigungen für diese Unterklassen definiert.

Ressourcentypen in Workload Manager:

WLM verwaltet drei Typen von Ressourcen auf der Basis des prozentualen Verbrauchs.

| Eintrag | Beschreibung |
|--|--|
| CPU-Auslastung durch Threads in einer Klasse | Dies ist Summe aller CPU-Zyklen, die von allen Threads in einer Klasse verbraucht wurden. |
| Auslastung des physischen Speichers durch die Prozesse in einer Klasse | Dies ist die Summe aller Speicherseiten, die zu den Prozessen in der Klasse gehören. |
| Platten-E/A-Bandbreite für die Klasse | Dies ist die Bandbreite (in 512-Byte-Blöcken pro Sekunde) aller E/As, die von Threads in der Klasse auf jeder Platteneinheit, auf die die Klasse zugreift, gestartet werden. |

WLM berechnet wie folgt im Sekundentakt die Auslastung pro Klasse für jede Ressource als Prozentsatz der insgesamt verfügbaren Ressource:

- Für die CPU entspricht die in jeder Sekunde verfügbare Gesamt-CPU-Zeit dem Produkt aus 1 Sekunde und der Anzahl der CPUs im System. Wenn beispielsweise auf einem 8-Wege-SMP-System alle Threads einer Klasse in der letzten Sekunde insgesamt 2 Sekunden der CPU-Zeit verbraucht haben, entspricht dies einem Prozentsatz von $2/8 = 25\%$. Der Prozentsatz, den WLM für die Regulierung verwendet wird, ist ein gewichteter Durchschnitt über einige Sekunden dieser "spontanen" sekundenbasierten Ressourcenauslastung.
- Bei physischem Hauptspeicher entspricht die jederzeit für Prozesse verfügbare Gesamtmenge der Gesamtanzahl der Speicherseiten, die physisch im System vorhanden sind, abzüglich der Anzahl festgehaltener Seiten. Festgehaltene Seiten werden von WLM nicht verwaltet, weil es nicht möglich ist, diese Seiten einer Klasse wegzunehmen und einer anderen Klasse zu geben, um die Speicherauslastung zu regulieren. Die Speicherauslastung einer Klasse ist das Verhältnis zwischen der Anzahl nicht festgehaltener Speicherseiten aller Prozesse in der Klasse und der Anzahl der verfügbaren Seiten im System, ausgedrückt in Prozent.
- Bei der Platten-E/A ist das Hauptproblem, die richtige verfügbare Bandbreite für eine Einheit zu bestimmen. Wenn eine Platte zu 100 % ausgelastet ist, unterscheidet sich ihr Durchsatz in Blöcken pro Sekunde, wenn eine Anwendung sequenzielle E/As durchführt, erheblich von dem Durchsatz, wenn mehrere Anwendungen zufällige E/As generieren. Falls Sie für die Berechnung der prozentualen Auslastung der Einheit bei zufälliger E/A nur den maximalen Durchsatz verwendet haben, der für sequenzielle E/A (als Wert der für die Einheit verfügbaren E/A-Bandbreite) gemessen wurde, könnten Sie irrtümlicherweise annehmen, dass die Einheit bei einer Auslastung von 20 % liegt, obwohl sie tatsächlich bei 100 % liegt.

Um genauere und zuverlässige Prozentwerte für die Plattenauslastung je Klasse zu erhalten, verwendet WLM die Statistiken, die von den Plattentreibern bereitgestellt (und mit dem Befehl **iostat** angezeigt) werden, die für jede Platteneinheit den prozentualen Zeitwert angeben, den die Einheit in der letzten Sekunde ausgelastet war. WLM zählt die Gesamtanzahl der Blöcke, die in der letzten Sekunde von allen Klassen, die auf die Einheit zugreifen, gelesen oder geschrieben wurden, die Anzahl der Blöcke, die von jeder einzelnen Klasse gelesen oder geschrieben wurde, und die prozentuale Auslastung der Einheit. Anschließend berechnet WLM den Prozentsatz des Plattendurchsatzes jeder einzelnen Klasse.

Wenn die Gesamtanzahl der während der letzten Sekunde gelesenen oder geschriebenen Blöcke beispielsweise 1000 und die Einheit zu 70 % ausgelastet ist, bedeutet dies, dass eine Klasse, die 100 Blöcke liest oder schreibt, 7 % der Plattenbandbreite nutzt. Wie bei der CPU-Zeit (einer weiteren erneuerbaren Ressource) sind die von WLM für die Regulierung der Platten-E/A verwendeten Werte gewichtete Durchschnittswerte über ein paar Sekunden dieser sekundenbasierten Prozentsätze.

Bei der Ressource Platten-E/A gelten die Anteile und Grenzwerte für jede Platteneinheit, auf die die Klasse zugreift. Die Regulierung wird für jede Einheit unabhängig durchgeführt. Das bedeutet, dass eine

Klasse auf einer Einheit über ihren Nutzungsrechten liegen kann und die E/As für diese Einheit reguliert werden, während sie auf einer anderen Platte unter ihren Nutzungsrechten liegt und die E/As für diese andere Einheit nicht beschränkt werden.

WLM unterstützt die Abrechnung und Regulierung von Ressourcen auf der Basis des Gesamtverbrauchs. Es gibt zwei Typen von Ressourcen, die auf diese Weise reguliert werden können: Klassengesamtwerte und Prozessgesamtwerte.

Klassengesamtwerte

Grenzwerte pro Klasse können für die Anzahl der Prozesse, Threads und Anmeldesitzungen in der Klasse definiert werden. Diese Grenzwerte werden für jede Ressource als absoluter Wert, der jederzeit in der Klasse vorhanden sein kann, angegeben. Diese Grenzwerte werden strikt eingehalten. Wenn eine Klasse ihren Grenzwert für eine dieser Klassen erreicht, scheitert jeder Versuch, eine weitere Instanz der Ressource zu erstellen. Die Operation schlägt so lange für jeden Prozess in der Klasse fehl, bis die Klasse wieder unter dem definierten Grenzwert für die Ressource liegt.

Prozessgesamtwerte

Grenzwerte pro Prozess können für die CPU-Gesamtzeit, die Anzahl der Blöcke für Platte-E/A und die Verbindungsdauer für eine Anmeldesitzung angegeben werden. Diese Grenzwerte werden auf Klassenebene definiert, gelten aber für jeden einzelnen Prozess in der Klasse (jeder Prozess, der diese Menge in Anspruch nehmen kann). Diese Verbrauchswerte sind kumulativ und stellen somit die Gesamtmenge jeder einzelnen Ressource dar, die von dem Prozess während seiner Lebensdauer verbraucht wird. Sobald ein Prozess seinen Gesamtgrenzwert für eine Ressource überschreitet, wird der Prozess beendet. Es wird ein Signal SIGTERM an den Prozess gesendet. Wenn der Prozess dieses Signal abfängt und nicht innerhalb einer Karenzzeit von 5 Sekunden beendet wird, wird ein Signal SIGKILL an den Prozess gesendet. Wenn eine Anmeldesitzung 90 % ihrer zulässigen Verbindungsdauer erreicht, wird eine Warnung am Steuerterminal ausgegeben, um den Benutzer darauf hinzuweisen, dass die Sitzung in Kürze beendet wird.

Zielanteile in Workload Manager:

Der (gewünschte) Zielressourcenverbrauch in Prozent für eine Klasse wird durch die Anzahl der Anteile bestimmt, die die Klasse für eine bestimmte Ressource hat.

Die Anteile geben an, wie viel einer bestimmten Ressource eine Klasse in Relation zu den anderen Klassen in ihrer Schicht erhalten soll. Der Zielprozentsatz einer Klasse für eine bestimmte Ressource ist nichts anderes als die Anzahl der Anteile, geteilt durch die Anzahl der aktiven Anteile in der Schicht der Klasse. Wenn darüber hinaus Grenzwerte verwendet werden, ist das Ziel auf den Bereich [Mindestwert, veränderlicher Maximalwert] beschränkt. Wenn das berechnete Ziel außerhalb dieses Bereichs liegt, wird es auf den entsprechenden oberen/unteren Grenzwert gesetzt (siehe Ressourcengrenzen). Die Anzahl aktiver Anteile entspricht der Gesamtanzahl der Anteile aller Klassen, die mindestens einen aktiven Prozess enthalten. Da die Anzahl aktiver Anteile dynamisch ist, gilt dies auch für das Ziel. Wenn eine Klasse die einzige aktive Klasse in einer Schicht ist, entspricht ihr Ziel 100 % der insgesamt für die Schicht verfügbaren Ressourcenmenge.

Angenommen, es gibt drei aktive Superklassen in den Schichten 0—A, B und C mit den Anteilen 15, 10 bzw. 5 für eine bestimmte Ressource. In diesem Fall ergeben sich die folgenden Ziele:

$$\text{Ziel(A)} = 15/30 = 50 \%$$

$$\text{Ziel(B)} = 10/30 = 33 \%$$

$$\text{Ziel(C)} = 5/30 = 17 \%$$

Wenn Klasse B zu einem späteren Zeitpunkt B aktiv wird (keine aktiven Prozesse enthält), werden die Ziele für die Klassen A und C automatisch angepasst:

$$\text{Ziel(A)} = 15/20 = 75 \%$$

$$\text{Ziel(C)} = 5/20 = 25 \%$$

Die Anteile stellen also einen sich selbst adaptierenden Prozentsatz dar. Sie ermöglichen eine gleichmäßige Verteilung der Ressourcen, die einer Klasse zugeordnet sind, auf die anderen Klassen, falls die Klasse inaktiv werden sollte, und eine Rückverteilung an die Klasse, sobald diese wieder aktiv ist.

Zur Unterstützung einer hohen Flexibilität kann für die Anzahl der Anteile für eine Klasse jeder Wert zwischen 1 und 65535 angegeben werden. Anteile können für Superklassen und Unterklassen angegeben werden. Für Superklassen sind die Anteile relativ zu allen anderen aktiven Klassen in derselben Schicht. Für Unterklassen sind die Anteile relativ zu allen anderen aktiven Unterklassen in derselben Superklasse in derselben Schicht. Die Anteile für eine Unterklasse einer Superklasse stehen in keiner Beziehung zu den Anteilen für eine Unterklasse einer anderen Superklasse.

Manchmal ist es wünschenswert, das Ziel für eine Klasse von der Anzahl aktiver Anteile unabhängig zu machen. In diesem Fall kann der Wert "-" für die Anzahl der Anteile angegeben werden. Die Regulierung der Klasse für diese Ressource wird daraufhin aufgehoben, d. h., die Klasse hat keine Anteile mehr, und ihr Ziel ist nicht von der Anzahl aktiver Anteile abhängig. Das Klassenziel wird berechnet, indem von der für die Schicht verfügbaren Ressourcenmenge die Summe der Mindestwerte für alle anderen Klassen in der Schicht abgezogen wird. Dieses Ziel bzw. der echte Verbrauch (je nachdem, welcher Wert geringer ist) wird anschließend von der Menge abgezogen, die den anderen Klassen in derselben Schicht zur Verfügung steht.

Angenommen, die Klassen A, B, C und D haben die folgenden Anteile für eine bestimmte Ressource: "-", 200, 150 bzw. 100. Alle Klassen sind aktiv, und Klasse A verbraucht 50 % der Ressource:

$$\begin{aligned}\text{Ziel(A)} &= \text{nicht reguliert} = 100 \% \\ \text{Ziel(B)} &= 200/450 * \text{verfügbar} = 44 \% * 50 \% = 22 \% \\ \text{Ziel(C)} &= 150/450 * \text{verfügbar} = 33 \% * 50 \% = 17 \% \\ \text{Ziel(D)} &= 100/450 * \text{verfügbar} = 22 \% * 50 \% = 11 \%\end{aligned}$$

Da Klasse A nicht reguliert ist und 50 % der verfügbaren Ressourcenmenge verbraucht, stehen den anderen Klassen nur 50 % zur Verfügung, und ihre Ziele werden basierend auf diesem Prozentsatz berechnet. Da Klasse A immer unter ihrem Ziel (100 %) liegt, hat sie immer eine höhere Priorität als alle anderen Klassen in derselben Schicht, die ihr Ziel erreicht bzw. überschritten haben. Weitere Informationen hierzu finden Sie im Abschnitt „Klassenpriorität in Workload Manager“ auf Seite 550.

Anmerkung: Die Regulierung einer Klasse für eine Ressource aufzuheben, ist nicht dasselbe, wie sie in eine höhere Schicht zu verschieben. Die im Folgenden aufgelisteten Verhaltensweisen gelten für eine nicht regulierte Klasse (in derselben Schicht), aber nicht, wenn die Klasse in eine höhere Schicht verschoben wird:

- Da die Anteile auf Ressourcenbasis definiert werden, kann die Regulierung einer Klasse für eine oder mehrere Ressourcen aufgehoben werden, während die Regulierung für andere Ressourcen bestehen bleibt.
- Die Mindestgrenzwerte für andere Klassen in derselben Schicht werden berücksichtigt. In höheren Schichten werden die in niedrigeren Schichten angegebenen Mindestwerte nicht berücksichtigt.
- Selbst wenn Mindestgrenzwerte für die Klassen mit Anteilen fehlen, ist der Verbrauch nicht regulierter Klassen in gewisser Weise von Klassen mit Anteilen abhängig, da sie um einen Teil der Ressource, die der Schicht zur Verfügung steht, konkurrieren. Es müssen einige Versuchsreihen durchgeführt werden, um festzustellen, welches Verhalten bei einer bestimmten Workload auftritt.

Wenn Sie die Anzahl der Anteile nicht angeben, wird der Standardwert "-" verwendet, und die Regulierung der Klasse für diese Ressource wird aufgehoben. In der ersten Version von WLM wurde 1 als der Standardwert für Anteile verwendet, wenn kein Wert angegeben war.

Die Anteile werden für alle Ressourcentypen pro Klasse in Zeilengruppen in der Datei **shares** angegeben. Beispiel:

```

shares
classname:
  CPU      = 2
  memory   = 4
  diskIO   = 3

```

Spezifikation von Ressourcengrenzen in Workload Manager:

Zusätzlich zur Definition von relativen Nutzungsrechten für Ressourcen mithilfe von Anteilen unterstützt WLM die Möglichkeit, Ressourcengrenzen für eine Klasse festzulegen. Ressourcengrenzen bieten dem Administrator eine bessere Kontrolle über die Ressourcenzuordnung. Die Grenzwerte werden als Prozentsätze, relativ zu der verfügbaren Ressourcenmenge der Schicht, zu der die Klasse gehört, angegeben.

Es gibt drei Typen von Grenzwerten für die auf Prozentsätzen basierende Regulierung:

Mindestwert

Dieser Wert gibt die Mindestressourcenmenge an, die der Klasse zur Verfügung gestellt werden muss. Wenn der tatsächliche Verbrauch der Klasse unter diesem Wert liegt, erhält die Klasse die höchste Zugriffspriorität für diese Ressource. Die gültigen Werte sind 0 bis 100, wobei 0 der Standardwert ist (wenn kein Wert angegeben wird).

Variabler Maximalwert

Dieser Wert gibt die maximale Ressourcenmenge an, die eine Klasse in Anspruch nehmen kann, wenn es zu einer Konkurrenzsituation um diese Ressource kommt. Wenn der Verbrauch der Klasse diesen Wert überschreitet, erhält die Klasse die niedrigste Priorität in ihrer Schicht. Sollte es keine Konkurrenzsituation um die Ressource (mit anderen Klassen derselben Schicht) geben, kann die Klasse so viel Ressourcen verbrauchen, wie sie möchte. Die gültigen Werte sind 1 bis 100, wobei 100 der Standardwert ist (wenn kein Wert angegeben wird).

Fester Maximalwert

Dieser Wert gibt die maximale Ressourcenmenge an, die eine Klasse in Anspruch nehmen kann, selbst wenn es zu keiner Konkurrenzsituation um diese Ressource kommt. Wenn die Klasse diesen Grenzwert erreicht, wird ihr der weitere Ressourcenverbrauch so lange untersagt, bis ihr prozentualer Verbrauch wieder unter den Grenzwert fällt. Die gültigen Werte sind 1 bis 100, wobei 100 der Standardwert ist (wenn kein Wert angegeben wird).

Die Werte für Ressourcengrenzen werden in der Ressourcengrenzendatei nach Ressourcentyp in Zeilengruppen für jede Klasse angegeben. Die Grenzwerte werden als Bereich mit dem Mindestwert als Anfangswert und dem flexiblen Maximalwert als Endwert mit einem Minuszeichen dazwischen angegeben. Leerzeichen werden ignoriert. Wenn Sie einen festen Maximalwert angeben möchten, muss dieser dem variablen Maximalwert folgen und von diesem durch ein Semikolon (;) getrennt werden. Jedem Grenzwert folgt ein Prozentzeichen (%).

Die folgenden Beispiele veranschaulichen die Verwendung der Regeldateien:

- Wenn der Benutzer joe aus der Gruppe acct3 den Befehl `/bin/vi` ausführt, wird der Prozess in die Superklasse acctg gestellt.
- Wenn der Benutzer sue aus der Gruppe dev den Befehl `/bin/emacs` ausführt, wird der Prozess in die Superklasse devlt gestellt (Übereinstimmung der Gruppen-ID), aber nicht der Unterklasse editors zugeordnet, weil Benutzer sue von dieser Klasse ausgeschlossen ist. Der Prozess wird standardmäßig in devlt gestellt.
- Wenn ein Datenbankadministrator `/usr/sbin/oracle` mit der Benutzer-ID oracle und der Gruppen-ID dbm für die Bereitstellung der Datenbank DB1 startet, wird der Prozess der Superklasse "default" zugeordnet. Nur wenn der Prozess seine Kennung auf `_DB1` setzt, wird er der Superklasse db1 zugeordnet.

Grenzwerte werden für alle Ressourcentypen nach Klasse in Zeilengruppen der Datei `limits` angegeben. Beispiel:

Anteile

Klassenname:

CPU = 0%-50%;80%
memory = 10%-30%;50%

In diesem Beispiel sind keine Grenzwerte für Platten-E/A definiert, deshalb werden die Systemstandardwerte verwendet. Daraus ergibt sich Folgendes:

diskIO = 0%-100%;100%

Die vorherigen Beispiele gehen davon aus, dass das Vererbungsattribut (inheritance) für die beschriebenen Superklassen und Unterklassen nicht aktiviert ist. Andernfalls würden die neuen Prozesse einfach die Superklasse oder Unterklasse von ihrem übergeordneten Prozess übernehmen.

Im Folgenden sind die einzigen Beschränkungen beschrieben, die WLM in Bezug auf die Werte für Ressourcengrenzen vorgibt:

- Der Mindestwert muss kleiner-gleich dem variablen Maximalwert sein.
- Der variable Maximalwert muss kleiner-gleich dem festen Maximalwert sein.
- Die Summe der Mindestwerte aller Superklassen in einer Schicht darf nicht größer als 100 sein.
- Die Summe der Mindestwerte aller Unterklassen einer bestimmten Superklasse darf nicht größer als 100 sein.

Wenn eine Klasse mit einem festen Speichergrenzwert diesen Grenzwert erreicht und weitere Seiten anfordert, wird der VMM-Algorithmus für Seitenersetzung (LRU) angestoßen, der der eingeschränkten Klasse Seiten "wegnimmt" (Page Steal) und damit die Seitenanzahl der Klasse unter den festen Maximalwert drückt, bevor er neue Seiten vergibt. Dieses Verhalten ist korrekt, aber die zusätzlichen Paging-Aktivitäten, die auch dann stattfinden, wenn viele freie Seiten verfügbar sind, wirken sich auf die allgemeine Leistung des Systems aus. Es wird empfohlen, vor dem Festlegen eines festen Maximalwerts für eine Klasse Mindestspeicherwerte für andere Klassen festzulegen.

Da Klassen, die unter ihrem Mindestwert liegen, die höchste Priorität in ihrer Schicht haben, sollte die Summe der Mindestwerte basierend auf den Ressourcenanforderungen der anderen Klassen in derselben Schicht auf einem angemessenen Niveau bleiben.

Die Einschränkung, dass die Summe der Mindestwerte innerhalb einer Schicht kleiner-gleich 100 sein muss, bedeutet, dass eine Klasse in der Schicht mit der höchsten Priorität immer Ressourcen bis zum definierten Mindestwert erhält. WLM garantiert nicht, dass die Klasse auch wirklich ihren Mindestwert erreicht. Dies richtet sich danach, wie die Prozesse in der Klasse ihre Ressourcen nutzen und welche anderen Grenzwerte wirksam sind. Es ist beispielsweise möglich, dass eine Klasse ihren Mindestwert für das CPU-Nutzungsrecht nicht erreichen, weil sie nicht genug Hauptspeicher erhält.

Das Festlegen eines Mindestspeichergrenzwerts für den physischen Speicher bietet einen gewissen Schutz für die Speicherseiten der Klassenprozesse (zumindest für die in der Schicht mit der höchsten Priorität). Einer Klasse sollten in dem Fall, dass sie unter ihrem Mindestwert liegt, nur dann Seiten "weggenommen" werden, wenn alle aktiven Klassen unter ihrem Mindestwert liegen und eine dieser Klassen mehr Seiten anfordert. Einer Klasse der höchsten Schicht sollten niemals Seiten "weggenommen" werden, wenn sie unter ihrem Mindestwert liegt. Das Festlegen eines Mindestspeicherwertes für eine Klasse mit interaktiven Jobs stellt sicher, dass dieser Klasse zwischen aufeinander folgenden Aktivierungen nicht alle Seiten weggenommen werden (selbst wenn der Speicher knapp ist), und verbessert die Antwortzeiten.

Achtung: Feste Maximalwerte können bei unangemessener Anwendung erheblichen Einfluss auf die System- bzw. Anwendungsleistung haben. Da das Festlegen fester Grenzwerte nicht genutzte Systemressourcen zur Folge haben kann, eignen sich variable Maximalwerte in den meisten Fällen besser. Feste Grenzwerte können beispielsweise verwendet werden, um den Verbrauch einer höheren Schicht einzuschränken, um einige Ressourcen einer niedrigeren Schicht zur Verfügung zu stellen. Das Umsiedeln von Anwendungen, die Ressourcen benötigen, in eine höhere Schicht ist jedoch unter Umständen die bessere Lösung.

Die Gesamtgrenzwerte können in der Datei "limits" mit den in der folgenden Tabelle zusammengefassten Werten und Einheiten festgelegt werden:

Tabelle 69. Ressourcengrenzen für Workload Manager

| Ressource | Zulässige Einheiten | Standardeinheit | Maximalwert | Mindestwert |
|------------------|---------------------|-----------------|------------------------------|-------------|
| totalCPU | s, m, h, d, w | s | $2^{30} - 1$ s | 10 s |
| totalDiskIO | KB, MB, TB, PB, EB | KB | $(2^{63} - 1) * 512/1024$ KB | 1 MB |
| totalConnectTime | s, m, h, d, w | s | $2^{63} - 1$ s | 5 m |
| totalProcesses | - | - | $2^{63} - 1$ | 2 |
| totalThreads | - | - | $2^{63} - 1$ | 2 |
| totalLogins | - | - | $2^{63} - 1$ | 1 |

Anmerkung: Bei den Einheitenangaben wird nicht zwischen Groß-/Kleinschreibung unterschieden. s steht für Sekunden, m für Minuten, h für Stunden, d für Tage, w für Wochen, KB für Kilobytes, MB für Megabytes usw.

Im Folgenden sehen Sie ein Beispiel für eine Zeilengruppe aus einer Datei limits:

```
BadUserClass:
    totalCPU = 1m
    totalConnectTime = 1h
```

Die Gesamtgrenzwerte können abgesehen von den folgenden Einschränkungen mit jedem Wert aus der obigen Tabelle angegeben werden:

- Wenn ein Wert für totalThreads angegeben wird, muss dieser größer-gleich dem Wert von totalProcesses sein.
- Wenn totalThreads angegeben wird und totalProcesses nicht, wird der Grenzwert für totalProcesses auf den Wert von totalThreads gesetzt.

Die Gesamtgrenzwerte können auf Superklassen- und Unterklassenebene angegeben werden. Wenn die Grenzwerte geprüft werden, wird der Grenzwert für die Unterklasse vor dem Grenzwert für die Superklasse geprüft. Sind beide Grenzwerte angegeben, wird der niedrigere der beiden wirksam. Wenn der für die Unterklasse angegebene Grenzwert größer ist als der Grenzwert für die zugehörige Superklasse, wird beim Laden der Konfiguration zwar eine Warnung ausgegeben, aber der Ladeprozess fortgesetzt. Dies ist für die Klassengesamtgrenzwerte entscheidend, weil der Grenzwert absolut (nicht relativ zur Superklasse) ist und eine Unterklasse alle Ressourcen verbrauchen könnte, die der Superklasse zur Verfügung stehen. Wenn dieser Wert nicht angegeben ist, ist der Standardwert für alle Gesamtgrenzwerte "-", was bedeutet, dass es keinen Grenzwert gibt. Die Gesamtabrechnung und die Regulierung für Klassen und Prozesse sind standardmäßig aktiviert, wenn WLM aktiv ist. Mit der Option **-T [class | proc]** des Befehls **wlmcntrl** können Gesamtabrechnung und Regulierung inaktiviert werden.

Klassenvorität in Workload Manager:

WLM ordnet Klassen Ressourcen zu, indem er jeder Klasse für jede Ressource eine Priorität zuweist.

Die Priorität einer Klasse ist dynamisch und basiert auf der Schicht, den Anteilen, Grenzwerten und dem aktuellen Verbrauch der Klasse. Den Klassen mit der höchsten Priorität wird stets ein bevorzugter Zugriff

auf die Ressourcen eingeräumt. Schichten der höchsten Ebene stellen nicht überlappende Bereiche von Klassenprioritäten dar. Die Klassen in der Schicht 0 erhalten immer eine höhere Priorität als Klassen in der Schicht 1 (sofern sie nicht über dem festen Maximalwert liegen).

Für die Festlegung der Klassenpriorität wendet WLM seine Einschränkungen in der folgenden Reihenfolge an, die absteigend nach Bedeutung sortiert ist:

Feste Grenzwerte

Wenn der Klassenverbrauch den festen Maximalwert für eine Ressource überschreitet, erhält die Klasse die niedrigste Priorität für die Ressource, die vergeben werden kann, und kann so lange nicht auf die Ressource zugreifen, bis ihr Verbrauch wieder unter diesen Grenzwert fällt.

Schicht

Wenn keine festen Grenzwerte angegeben sind, wird die Priorität einer Klasse nach den zulässigen Mindest- und Maximalprioritätswerten für die Schicht der Klasse vergeben.

Veränderliche Grenzwerte

Wenn der Klassenverbrauch unter dem Mindestwert der veränderlichen Maximalwerte für eine Ressource liegt, erhält die Klasse die höchste Priorität in der Schicht. Liegt der Klassenverbrauch über dem variablen Maximalwert, erhält die Klasse die niedrigste Priorität in der Schicht.

Anteile

Mit Anteilen werden die Klassenverbrauchsziele für jede Ressource berechnet. Die Klassenpriorität wird herabgesetzt, wenn der Klassenverbrauch unter den Zielwert fällt, und erhöht, wenn der Klassenverbrauch den Zielwert überschreitet. Veränderliche Grenzwerte haben eine Vorrangstellung. Wenn veränderliche Grenzwerte angegeben sind, wird die Priorität der Klasse auf der Basis dieser veränderlichen Grenzwerte bestimmt.

Obwohl Anteile und Grenzwerte für jede Klasse und für jede Ressource verwendet werden können, sind die Ergebnisse vorhersehbarer, wenn jeweils nur eines dieser Kriterien pro Klasse verwendet wird.

Ressourcengruppen mit Prozessoren für exklusive Benutzung (XRSET):

Ressourcengruppen mit Prozessoren für exklusive Benutzung (XRSET, Exclusive Use Processor Resource Sets) ermöglichen Administratoren, Ressourcen für wichtige Arbeit vorzuhalten. Ein XRSET ist eine benannte Ressourcengruppe, die das Verhalten aller CPUs ändert, die in ihr enthalten sind. Sobald eine CPU als exklusiv gekennzeichnet ist, führt sie nur Programme aus, die direkt an sie weitergeleitet werden.

Ein XRSET erstellen

Ein XRSET kann nur von Root erstellt werden. Verwenden Sie den Befehl **mkrset**, um eine Ressourcengruppe im Namespace **sysxrset** zu erstellen. Der Befehl **mkrset -c 1-3 sysxrset/set1** erstellt beispielsweise ein XRSET für die CPUs 1, 2 und 3. Die Subroutine **rs_registername()** kann auch zum Erstellen eines XRSET verwendet werden.

Feststellen, ob XRSETs auf einem System definiert sind

Der Befehl **lsrset -v -n sysxrset** zeigt alle XRSETs an, die auf einem System definiert sind. (Derzeit gibt es keine API hierfür.)

Ein XRSET löschen

Ein XRSET kann nur von Root gelöscht werden. Der Befehl **rnrset** löscht ein XRSET. Die Subroutine **rs_discardname()** kann auch zum Löschen eines XRSET verwendet werden.

Warmstart des Systems durchführen

Wenn Sie einen Warmstart des Systems durchführen, werden alle definierten XRSETs aus der Registrierungsdatabank entfernt und sind nicht mehr wirksam.

Arbeit für XRSETs angeben

Es gibt mehrere Möglichkeiten, Arbeit für die Zuordnung zu Prozessoren für exklusive Benutzung zu kennzeichnen. Mit den Befehlen **attachrset** und **execrset** können Ressourcengruppen angegeben werden, die Prozessoren für exklusive Benutzung enthalten. Ressourcengruppen, die Prozessoren für exklusive Benutzung enthalten, können WLM-Klassen zugeordnet werden. Für die in solchen WLM-Klassen klassifizierte Arbeit werden die in der Ressourcengruppe angegebenen Prozessoren für exklusive Benutzung verwendet.

XRSETs mit `bindprocessor` und `_system_configuration.ncpus` verwenden

bindprocessor kann nicht verwendet werden, um zu veranlassen, dass Arbeit in Prozessoren für exklusive Benutzung ausgeführt wird. Hierfür können nur ressourcengruppenbasierte Zuordnungen verwendet werden.

Die Anzahl der CPUs in der Systemkonfiguration (Feld `_system_configuration.ncpus`) wird beim Erstellen von XRSETs nicht geändert. Es erscheinen trotzdem alle CPUs (NCPUs) im System.

Wenn Programme den Systemaufruf **bindprocessor** für NCPUs verwenden, scheitern CPUs in XRSETs mit dem Fehler EINVALID. Sie können eine Bindung an jede ID vornehmen, die von der Abfrageoption des Befehls **bindprocessor** zurückgegeben wird. Die Abfrageoption (**bindprocessor -q**) gibt nur gültige BIND-IDs zurück. Ausgeschlossen davon sind die IDs, die exklusiven CPUs zugeordnet sind.

Wenn beispielsweise 10 CPUs in einem System online sind und drei davon in XRSETs enthalten sind, ist der Aufruf **bindprocessor** an CPUs mit BIND-IDs aus dem Bereich 0 bis 6 erfolgreich. Ein Aufruf **bindprocessor** an CPUs mit BIND-IDs aus dem Bereich 7 bis 9 generiert einen Fehler des Typs EINVALID.

XRSETs mit Operationen für dynamische Rekonfiguration von CPUs verwenden

Im Allgemeinen ist die dynamische Rekonfiguration von CPUs von Prozessoren für exklusive Benutzung nicht betroffen. Die Erstellung von XRSETs und die Zuordnung von Arbeit zu diesen Prozessen können das Entfernen einer CPU verhindern. CPUs, die dem System dynamisch hinzugefügt werden, können Prozessoren für allgemeine Benutzung oder Prozessoren für exklusive Benutzung sein. Sie sind Prozessoren für exklusive Benutzung, wenn ein XRSET vorhanden ist, das die entsprechende logische CPU-ID enthält.

Ressourcengruppen in Workload Manager:

WLM verwendet Ressourcengruppen (oder *rsets*), um die Prozesse in einer bestimmten Klasse auf eine Untergruppe der physischen Ressourcen des Systems zu beschränken. In WLM gehören der Hauptspeicher und die Prozessoren zu den verwalteten physischen Ressourcen. Eine gültige Ressourcengruppe setzt sich aus Hauptspeicher und mindestens einem Prozessor zusammen.

Mit SMIT kann ein Systemadministrator Ressourcengruppen definieren und benennen, die eine Untergruppe der im System verfügbaren Ressourcen enthalten. Anschließend kann Root oder ein ernannter Administrator der Superklasse mit den WLM-Verwaltungsschnittstellen den Namen der Ressourcengruppe als Attribut **rset** einer WLM-Klasse verwenden. Danach werden alle Prozesse, die dieser WLM-Klasse zugeordnet sind, nur noch an einen der Prozessoren in der Ressourcengruppe verteilt, wodurch die Workloads für die CPU-Ressource effektiv getrennt werden.

Alle aktuellen Systeme haben nur eine Speicherdomäne, die von allen Ressourcengruppen gemeinsam genutzt wird, so dass die Workloads mit dieser Methode im Hauptspeicher nicht physisch getrennt werden.

Registry für Ressourcengruppen in Workload Manager:

Mit den Registry-Services für Ressourcengruppen (**rset**) können Systemadministratoren Ressourcengruppen definieren und benennen, so dass sie von anderen Benutzern oder Anwendungen verwendet werden können.

Zur Minderung des Risikos von Namenskollisionen unterstützt die Registry ein zweistufiges Benennungsschema. Der Name einer Ressourcengruppe hat das Format *Namespace/rset-Name*. *Namespace* und *rset-Name* können jeweils 255 Zeichen lang sein und dürfen nur Groß- und Kleinbuchstaben, Zahlen, Unterstreichungszeichen und Punkte (.) enthalten. Außerdem muss die Groß-/Kleinschreibung beachtet werden. Der *Namespace sys* ist für das Betriebssystem reserviert und wird für *rset*-Definitionen verwendet, die die Ressourcen im System darstellen.

Die Namen von *rset*-Definitionen sind innerhalb des Registry-Namespace eindeutig. Wenn Sie der Registry eine neue *rset*-Definition mit einem Namen hinzufügen, der einer bereits vorhandenen *rset*-Definition entspricht, wird die vorhandene Definition durch die neue Definition ersetzt, sofern die erforderlichen Berechtigungen vorhanden sind. Nur Root kann mit SMIT Ressourcengruppen erstellen, ändern und löschen sowie die integrierte *rset*-Datenbank aktualisieren.

Jede *rset*-Definition hat einen Eigner (Benutzer-ID), eine Gruppe (Gruppen-ID) und Zugriffsberechtigungen. Diese werden beim Erstellen der *rset*-Definition angegeben und werden für die Zugriffssteuerung verwendet. Wie bei Dateien gibt es separate Zugriffsberechtigungen für Eigner, Gruppe und andere Benutzer, die definieren, ob Lese- und/oder Schreibberechtigung erteilt wird. Mit Leseberechtigung kann eine *rset*-Definition abgerufen werden. Mit Schreibberechtigung kann eine *rset*-Definition geändert oder entfernt werden.

Die von den Systemadministratoren definierten *rset*-Definitionen werden in der Zeilengruppendatei */etc/rsets* verwaltet. Das Format dieser Datei wird nicht beschrieben. Benutzer müssen die Ressourcengruppe (*rset*) mit der SMIT-Schnittstelle bearbeiten, um potenzielle Kompatibilitätsprobleme zu vermeiden, wenn das Dateiformat geändert wird. Wie bei WLM-Klassendefinitionen müssen *rset*-Definitionen in die Datenstrukturen des Kernels geladen werden, damit sie von WLM verwendet werden können.

Zugehörige Tasks:

„Eine Ressourcengruppe erstellen“ auf Seite 526

Die Verwendung von Ressourcengruppen (*rsets*, Resource Sets) ist eine effektive Methode, um Workloads voneinander abzugrenzen, soweit es die CPU betrifft. Indem Sie zwei unterschiedliche Workloads in zwei Klassen trennen und jeder Klasse eine andere Untergruppe der CPUs zuteilen, können Sie sicherstellen, dass die beiden Workloads nicht um CPU-Ressourcen konkurrieren. Eine Konkurrenzsituation um physischen Speicher und E/A-Bandbreite ist jedoch damit nicht ausgeschlossen.

Workload Manager konfigurieren

Klassendefinitionen, Klassenattribute, Anteile und Grenzwerte sowie die Regeln für die automatische Klassenzuordnung können mit SMIT oder der WLM-Befehlszeilenschnittstelle eingegeben werden. Diese Definitionen und Regeln werden in unverschlüsselten Textdateien gespeichert, die auch mit einem Texteditor erstellt oder geändert werden können.

Diese Dateien (die so genannten *WLM-Eigenschaftendateien*) werden in den Unterverzeichnissen des Verzeichnisses */etc/wlm* gespeichert. Eine WLM-Konfiguration wird mit einer Gruppe von Dateien, die die Superklassen und ihre zugehörigen Unterklassen beschreiben, definiert. Die Dateien für die WLM-Konfiguration **Config** werden im Verzeichnis */etc/wlm/Config* gespeichert. Dieses Verzeichnis enthält die Definitionen der WLM-Parameter für die Superklassen. Die Dateien haben die Namen *description*, *classes*, *shares*, *limits* und *rules*. Dieses Verzeichnis kann außerdem Unterverzeichnisse enthalten, die den Namen der Superklassen entsprechen und in denen die Definitionen der Unterklassen gespeichert werden. Für die Superklasse *Super* der WLM-Konfiguration **Config** enthält das Verzeichnis */etc/wlm/Config/Super* beispielsweise die Eigenschaftendateien für die Unterklassen der Superklasse *Super*. Die Dateien haben die Namen *description*, *classes*, *shares*, *limits* und *rules*.

Nachdem eine WLM-Konfiguration vom Systemadministrator definiert wurde, kann diese mit dem Direktaufruf **smit wlmmanage** oder mit dem Befehl **wlmcntrl** als aktive Konfiguration festgelegt werden.

Sie können mehrere Gruppen von Eigenschaftendateien definieren, die jeweils eine andere Konfiguration des Workload-Managements definieren. Diese Konfigurationen werden gewöhnlich in Unterverzeichnissen des Verzeichnisses */etc/wlm* gespeichert. Die symbolische Verbindung */etc/wlm/current* zeigt auf das

Verzeichnis, das die aktuellen Konfigurationsdateien enthält. Diese Verbindung wird vom Befehl **wlmcntrl** aktualisiert, wenn WLM mit einer bestimmten Gruppe von Konfigurationsdateien gestartet wird.

Zugehörige Informationen:

„Workload Manager für die Konsolidierung von Workloads konfigurieren“ auf Seite 529

Mit Workload Manager (WLM) können Sie steuern, welche Ressourcen von Jobs auf Ihrem System verwendet werden.

Anwendungsanforderungen für die Konfiguration von Workload Manager:

Für die erste Phase einer Konfigurationsdefinition ist es erforderlich, dass Sie Ihre Benutzer und ihre Anforderungen an die Datenverarbeitung sowie die Anwendungen auf Ihrem System, ihre Ressourcenanforderungen und die Anforderungen Ihres Geschäfts (z. B. welche Tasks kritisch sind und welchen eine geringere Priorität gegeben werden kann) verstehen. Basierend auf diesem Verständnis definieren Sie Ihre Superklassen und anschließend Ihre Unterklassen.

Die Festlegung der Prioritäten richtet sich danach, welche Funktion WLM in Ihrer Organisation einnimmt. Bei einer Serverkonsolidierung kennen Sie unter Umständen die Anwendungen, Benutzer und ihre Ressourcenanforderungen und sind in der Lage, einige der Schritte zu überspringen oder zu verkürzen.

Mit WLM können Sie Prozesse nach Benutzer oder Gruppe, Anwendung, Typ, Kennung oder einer Kombination dieser Attribute klassifizieren. Da WLM die Ressourcenauslastung zwischen den Klassen reguliert, sollten Systemadministratoren Anwendungen und Benutzer nach denselben Ressourcenauslastungsmustern in dieselben Klassen gruppieren. Sie können beispielsweise die interaktiven Jobs, die in der Regel wenig CPU-Zeit beanspruchen, aber schnelle Antwortzeiten erfordern, von Stapeljobs trennen, die gewöhnlich sehr CPU- und speicherintensiv sind. Dasselbe gilt für eine Datenbankumgebung, in der Sie unter Umständen OLTP-Datenverkehr von den rechenintensiven Anfragen des Data-Minings trennen müssen.

Dieser Schritt wird mit SMIT oder der Befehlszeilenschnittstelle ausgeführt. Für die ersten Male empfiehlt es sich wahrscheinlich, SMIT zu verwenden, die Sie durch die Schritte beim Erstellen Ihrer ersten WLM-Konfiguration, einschließlich Definition der Superklassen und ihrer Attribute führen. Im ersten Durchlauf können Sie einige Attribute definieren und für andere den Standardwert übernehmen. Dasselbe gilt für Ressourcenanteile und -grenzen. Alle diese Klassenmerkmale können zu einem späteren Zeitpunkt dynamisch geändert werden.

Sie können WLM anschließend im Passivmodus starten, Ihre Klassifizierung überprüfen und beginnen, die Ressourcenauslastungsmuster Ihrer Anwendungen zu überprüfen.

Überprüfen Sie Ihre Konfiguration mit dem Befehl **wlmcheck** oder mit den entsprechenden SMIT-Menüs. Starten Sie WLM anschließend mit der neu definierten Konfiguration im Passivmodus. WLM klassifiziert alle vorhandenen Prozesse (und alle Prozesse, die ab diesem Zeitpunkt gestartet werden) und beginnt mit der Zusammenstellung von Statistiken zur CPU-, Speicher- und Platten-E/A-Auslastung der verschiedenen Klassen. Eine Regulierung dieser Ressourcennutzung wird von WLM nicht durchgeführt.

Vergewissern Sie sich, dass die verschiedenen Prozesse den gemäß Systemadministratorvorgaben richtigen Klassen zugeordnet sind. Verwenden Sie hierfür das Flag **-o** des Befehls **ps**. Wenn einige der Prozesse nicht wie erwartet klassifiziert wurden, können Sie Ihre Zuordnungsregeln ändern oder den Vererbungsparameter (inheritance) für einige der Klassen definieren, wenn Sie möchten, dass die neuen Prozesse in derselben Klasse verbleiben wie der jeweils übergeordnete Prozess, und WLM anschließend aktualisieren. Dieser Prozess kann so oft wiederholt werden, bis Sie mit dieser ersten Stufe der Klassifizierung (Superklassen) zufrieden sind.

Die Ausführung von WLM im Passivmodus und die Aktualisierung von WLM (immer im Passivmodus) bergen nur geringe Risiken, erzeugen nur geringen zusätzlichen Aufwand und können auf einem Pro-

duktionssystem sicher vorgenommen werden, ohne den normalen Systembetrieb zu beeinträchtigen. Verwenden Sie zum Aktivieren und Aktualisieren von WLM den Befehl **wlmcntrl**, den Sie entweder in der Befehlszeile oder über SMIT aufrufen können.

Führen Sie WLM im Passivmodus aus, um mit dem Befehl **wlmstat** Statistiken zu erfassen. Der Befehl **wlmstat** kann in regelmäßigen Intervallen verwendet werden, um die Ressourcenauslastung pro Klasse als Prozentsatz der insgesamt verfügbaren Ressourcen für die Superklassen anzuzeigen. Auf diese Weise können Sie Ihr System über längere Zeiträume überwachen, um die Ressourcenauslastung Ihrer wichtigsten Anwendungen zu prüfen.

Schichten, Anteile und Grenzwerte in Workload Manager:

Entscheiden Sie auf der Basis der Daten, die Sie während der Ausführung von WLM im passiven Modus erfasst haben, und Ihrer Geschäftsziele, welche Schichtnummer den einzelnen Superklassen und welche Anteile jeder Ressource den verschiedenen Klassen zugeordnet werden sollen.

Für einige Klassen können Sie Mindest- und Maximalwerte definieren. Passen Sie die Anteile und Schichtnummern an, um Ihre Ressourcenzuordnungsziele zu erreichen. Behalten Sie Grenzwerte für solche Fälle vor, die mit Anteilen allein nicht gelöst werden können. Möglicherweise stellen Sie auch fest, dass Sie Unterklassen hinzufügen müssen.

- Verwenden Sie Mindestwerte für Anwendungen, die gewöhnlich eine geringe Ressourcennutzung haben, aber kurze Antwortzeiten benötigen, wenn sie durch ein externes Ereignis aktiviert werden. Eines der Probleme, die interaktive Jobs in Situationen haben, in denen der Hauptspeicher knapp wird, besteht darin, dass ihnen die Seiten "weggenommen" werden, während sie inaktiv sind. Durch einen Mindestwert können einige der Seiten interaktiver Jobs geschützt werden, wenn sich die Klasse in der Schicht 0 befindet.
- Verwenden Sie Maximalwerte für einige ressourcenintensive Jobs mit niedriger Priorität. Sofern Sie Ihre Systemressourcen nicht aus anderen Gründen aufteilen, ist ein fester Maximalwert in den meisten Fällen nur für eine nicht erneuerbare Ressource wie den Hauptspeicher geeignet. Der Grund hierfür ist die Zeit, die benötigt wird, um Daten in den Paging-Bereich zu schreiben, wenn eine Klasse mit einer höheren Priorität Seiten benötigt, die die erste Klasse verwendet hätte. Für die CPU-Belastung können Sie Schichten oder veränderliche Maximalwerte verwenden, um sicherzustellen, dass einer Klasse mit höherer Priorität unverzüglich CPU-Zeit zugeordnet wird.

Beim Erstellen und Anpassen der Parameter von Unterklassen bis zu einem zufriedenstellenden Systemverhalten können Sie WLM nur für die Unterklassen einer bestimmten Superklasse aktualisieren, die keine Auswirkung auf Benutzer und Anwendungen in den anderen Superklassen haben.

Je nach geschäftlichen Anforderungen können Sie auch andere Konfigurationen mit unterschiedlichen Parametern definieren. In diesem Fall können Sie durch Kopieren und Ändern vorhandener Konfigurationen Zeit einsparen.

Konfiguration von Workload Manager optimieren:

Überwachen Sie das System mit dem Befehl **wlmstat**, und stellen Sie sicher, dass die von WLM vorgenommene Regulierung Ihren Zielen entspricht und nicht einigen Anwendungen übermäßig vieler Ressourcen "beraubt", während andere Anwendungen mehr Ressourcen erhalten, als sie eigentlich sollten. In einem solchen Fall müssen Sie die Anteile anpassen und WLM aktualisieren.

Während Sie die Werte für die Anteile, Grenzwerte und Schichten überwachen, entscheiden Sie, ob Sie die Verwaltung der Unterklassen für einige oder alle Superklassen delegieren möchten. Der Administrator kann dann die Werte für die Anteile, Grenzwerte und Schichten überwachen und konfigurieren.

Der Administrator jeder Superklasse kann diesen Prozess für die Unterklassen jeder Superklasse wiederholen. Der einzige Unterschied besteht darin, dass WLM auf Ebene der Unterklassen nicht im passiven Modus ausgeführt werden kann. Die Konfiguration und Optimierung von Unterklassen muss vorgenom-

men werden, wenn WLM im aktiven Modus arbeitet. Eine Möglichkeit, die Benutzer und Anwendungen in der Klasse nicht zu beeinträchtigen, besteht darin, die jeweilige Schicht zu starten und die Anteile und Grenzwerte für die Unterklassen mit den Standardwerten ('-' (Minuszeichen) für Anteile, 0 % für Mindestwert und 100 % für veränderlichen und festen Maximalwert) zu aktivieren. Mit diesen Einstellungen findet keine Regulierung der Ressourcenzuordnung für die Unterklassen durch WLM statt.

Fehlerbehebung beim Workload Manager

Wenn Ihre aktuelle Konfiguration nicht das gewünschte Verhalten aufweist, müssen Sie unter Umständen Ihre WLM-Konfiguration anpassen.

Die Einsatzwerte für jede Klasse können mit dem Befehl **wlmstat** überwacht werden. Diese Daten können erfasst und analysiert werden, um festzustellen, welche Änderungen unter Umständen an der Konfiguration vorgenommen werden müssen. Nach der Aktualisierung der Konfiguration aktualisieren Sie die aktive WLM-Konfiguration mit dem Befehl **wlmcntrl -u**.

Die folgenden Richtlinien können Sie beim Ändern Ihrer Konfiguration unterstützen:

- Wenn die Anzahl aktiver Anteile in einer Schicht mit der Zeit stark variiert, können Sie entscheiden, einer Klasse keine Anteile für eine Ressource zu geben, so dass sie ein Verbrauchsziel haben kann, das von der Anzahl aktiver Anteile unabhängig ist. Dieses Verfahren ist hilfreich für wichtige Klassen, die bevorzugten Zugriff auf eine Ressource benötigen.
- Wenn Sie den Zugriff auf eine bestimmte Menge der Ressource garantieren müssen, legen Sie Mindestwerte fest. Dieses Verfahren ist hilfreich für interaktive Jobs, die nicht viele Ressourcen verbrauchen, aber schnell auf externe Ereignisse reagieren müssen.
- Wenn Sie den Zugriff auf Ressourcen beschränken müssen, aber mit Anteilen keine ausreichende Steuerung erzielen können, geben Sie Maximalwerte an. In den meisten Fällen sind veränderliche Maximalwerte ausreichend. Für eine strikte Umsetzung können jedoch feste Maximalwerte verwendet werden. Da feste Maximalwerte einer Verschwendung von Systemressourcen zur Folge haben und Paging-Bereichsaktivitäten erhöhen können, wenn sie für die Speicherregulierung verwendet werden, sollten Sie für die anderen Klassen Mindestwerte festlegen, bevor Sie irgendwelche festen Grenzwerte definieren.
- Wenn weniger wichtige Jobs wichtigere Jobs stören, ordnen Sie erstere einer niedrigeren Schicht zu. Dieses Verfahren gewährleistet, dass weniger wichtige Jobs eine niedrigere Priorität haben und nicht mit wichtigeren aktiven Jobs um verfügbare Ressourcen konkurrieren.
- Wenn eine Klasse ihr Verbrauchsziel für eine Ressource nicht erreichen kann, prüfen Sie, ob diese Bedingung durch Konkurrenzsituationen für eine andere Ressource verursacht wird. Wenn ja, ändern Sie die Klassenzuordnung für die Ressource, um die es in der Konkurrenzsituation geht.
- Wenn Prozesse in einer Klasse in ihren Verhaltensweisen oder im Ressourcenverbrauch stark variieren, erstellen Sie weitere Klassen, um eine differenziertere Steuerung zu haben. Außerdem könnte es sich empfehlen, für jede wichtige Anwendung eine separate Klasse zu erstellen.
- Wenn Ihre Analyse zeigt, dass die von einer Klasse benötigte Ressource von dem Verbrauch einer anderen Klasse abhängig ist, ändern Sie die Zuordnung Ihrer Ressourcen entsprechend. Wenn beispielsweise die von KlasseZ benötigte Ressourcenmenge von der Anzahl an Verarbeitungsaufträgen abhängig ist, die von KlasseA verarbeitet werden können, muss KlasseA der Zugriff auf einen so großen Teil der Ressource garantiert werden, dass sie bereitstellen kann, was KlasseZ benötigt.
- Wenn eine oder mehrere Anwendungen ständig nicht genügend Ressourcen erhalten, um eine angemessene Leistung erbringen zu können, haben Sie unter Umständen nur die Möglichkeit, die Workload auf dem System zu verringern.

Anmerkung: Sie können einen *Benutzer mit Administratorberechtigung* für eine Superklasse definieren, um den Arbeitsumfang des WLM-Administrators zu reduzieren. Nachdem die Konfiguration der Ausgangsebene getestet und optimiert wurde, können anschließend von den Benutzern mit Administratorberechtigung für Superklassen Anpassungen an ihre jeweiligen Anforderungen vorgenommen werden (z. B. Unterklassen erstellen und konfigurieren).

Workload-Manager-API

Anwendungen können die WLM-APIs, eine Gruppe von Routinen in der Bibliothek `/usr/lib/libwlm.a`, verwenden, um alle Tasks auszuführen, die ein WLM-Administrator mit der Befehlszeilenschnittstelle von WLM ausführen kann.

Dazu gehören die folgenden Tasks:

- Klassen erstellen, ändern oder löschen
- Klassenattribute oder Ressourcenanteile und -grenzwerte ändern
- Klassen entfernen
- Prozesse manuell zu Klassen zuordnen
- WLM-Statistiken abrufen

Die API ermöglicht Anwendungen, ein anwendungsspezifisches Klassifizierungsattribut, eine so genannte Kennung, zu setzen. Wenn diese Kennung auf der Basis einer Gruppe von Werten, die der Systemadministrator (in der Benutzerdokumentation zur Anwendung) bereitstellt, festgelegt wird, ist es möglich, zwischen mehreren Instanzen derselben Anwendung zu unterscheiden. Die unterschiedlichen Klassen können somit mit unterschiedlichen Nutzungsrechten für Ressourcen klassifiziert werden.

Außerdem ermöglicht die Routine `wlm_set_tag` einer Anwendung, eine Anwendungskennung festzulegen und anzugeben, ob diese Kennung von untergeordneten Prozesse, die mit `fork` oder `exec` erzeugt werden, übernommen werden soll. Anwendungskennungen können Threads auch mit der Anweisung `wlm_set_thread` zugeordnet werden. Die Anwendungskennung eines Thread kann von untergeordneten Threads, die mit den Subroutinen `fork`, `exec` oder `pthread_create` erzeugt werden, übernommen werden. Die Bibliothek bietet Unterstützung für 32-Bit- und 64-Bit-Multithread-Anwendungen.

Anwendungskennung:

Die Anwendungskennung ist eine Zeichenfolge und wird als eines der Klassifizierungskriterien für die automatische Klassifizierung von Prozessen oder Threads (mit der Datei `rules`) verwendet. Diese Kennung ist im Grunde ein anwendungsdefiniertes Klassifizierungskriterium, das zusätzlich zu den systemdefinierten Kriterien wie `user`, `group`, `application` und `type` verwendet wird.

Wenn ein Anwendungsprozess oder Thread seine Kennung setzt, wird er sofort unter Verwendung der geltenden Superklassen- und Unterklassenregeln für die derzeit aktive WLM-Konfiguration neu klassifiziert. WLM überprüft die Zuordnungsregeln und sucht unter Verwendung aller Prozessattribute, einschließlich der neuen Kennung nach einer Übereinstimmung.

Die Kennung muss, um wirksam zu sein, in mindestens einer der Zuordnungsregeln verwendet werden. Das Format und die Verwendung der verschiedenen Kennungen durch die einzelnen Anwendungen müssen in der Verwaltungsdokumentation der Anwendung eindeutig angegeben sein und den WLM-Administratoren bekannt sein, so dass sie die verschiedenen Werte der Kennungen in ihren Zuordnungsregeln verwenden können, um zwischen verschiedenen Instanzen derselben Anwendung zu unterscheiden.

Da unterschiedliche Benutzer unterschiedliche Anforderungen in Bezug auf die Merkmalgruppe für Anwendungsprozesse haben können, die sie für die Klassifizierung dieser Prozesse verwenden möchten, wird empfohlen, dass die Anwendung eine Gruppe von Konfigurations- oder Laufzeitattributen bereitstellt, die für die Erstellung der Kennung verwendet werden können. Der Anwendungsadministrator kann das Format der Kennung für die Anwendung festlegen. Die Attribute, die für die Kennung verwendet werden können, und die Syntax für das Format der WLM-Kennung sind anwendungsabhängig und liegen in der Verantwortung des Anwendungsproviders.

Eine Instanz eines Datenbankservers kann beispielsweise ermitteln, mit welcher Datenbank sie arbeitet (`db_name`) und über welchen TCP-Port ein bestimmter Benutzer verbunden wird (`port_num`). Administratoren können folgende Prioritäten haben:

- Jeder Klasse unterschiedliche Nutzungsrechte für Ressourcen erteilen, um unterschiedliche Klassen für Prozesse zu erstellen, die auf unterschiedliche Datenbanken zugreifen.
- Prozesse, die ferne Anforderungen unterschiedlichen Ursprungs bearbeiten, unterscheiden und die Portnummer als Klassifizierungsattribut verwenden.
- Eine Superklasse für jede Datenbank und eine Unterklasse für jede Portnummer jeder Superklasse erstellen.

Eine Möglichkeit, diese unterschiedlichen Anforderungen zu konsolidieren, ist die Spezifikation von Inhalt und Format der Kennung. Angenommen, die Kennung kann in einer Konfigurationsdatei oder in einem Laufzeitparameter wie `WLM_TAG=$db_name` oder `WLM_TAG=$db_name_$port_num` an die Anwendung übergeben werden.

Wenn sie ihre Kennung setzt, kann eine Anwendung angeben, ob diese Kennung von allen untergeordneten Anwendungen übernommen wird, so dass alle Prozesse, die von einer bestimmten Instanz einer Anwendung erzeugt werden, derselben Klasse zugeordnet werden können. Die Anwendungskennung wird sehr häufig im Zusammenhang mit dem Attribut "inheritance" verwendet.

Das folgende Beispiel zeigt, wie Anwendungskennungen verwendet werden können. In diesem Beispiel ist die Kennung der Datenbank mit dem Datenbanknamen identisch. Wenn zwei Instanzen des Servers mit zwei unterschiedlichen Datenbanken arbeiten, werden zwei unterschiedliche Kennungen definiert, z. B. db1 und db2.

Ein Systemadministrator kann zwei unterschiedliche Klassen, dbserv1 und dbserv2 erstellen, und die beiden Datenbankserver (und alle untergeordneten Prozesse, sofern der Parameter inheritance verwendet wird) anhand der Kennungen diesen Klassen zuordnen. In diesem Fall ist es möglich, jeder Klasse auf der Basis bestimmter Geschäftsziele unterschiedliche Nutzungsrechte für Ressourcen zu erteilen.

Die entsprechenden Zuordnungsregeln könnten wie folgt aussehen:

```
* class  resvd  user  group  application  type  tag
*
dbserv1  -    -    dbadm  /usr/sbin/dbserv  -    db1
dbserv2  -    -    dbadm  /usr/sbin/dbserv  -    db2
```

API-Typen:

Im Folgenden werden die API-Typen von Workload Manager (WLM) beschrieben.

APIs für die Klassenverwaltung

Die Anwendungsprogrammierschnittstelle (API, Application Programming Interface) von WLM ermöglicht Anwendungen Folgendes:

- Namen und Merkmale der vorhandenen Klassen einer bestimmten WLM-Konfiguration abfragen (**wlm_read_classes**)
- Eine neue Klasse für eine bestimmte WLM-Konfiguration erstellen, die Werte der verschiedenen Klassenattribute (z. B. tier (Schicht) und inheritance (Vererbung)) sowie die Anteile und Grenzwerte für die von WLM verwalteten Ressourcen (z. B. CPU, physischer Speicher und blockorientierte Ein-/Ausgabe) definieren (**wlm_create_class**)
- Merkmale einer vorhandenen Klasse einer bestimmten WLM-Konfiguration ändern, einschließlich der Klassenattribute und Anteile und Grenzwerte für Ressourcen (**wlm_change_class**)
- Eine vorhandene Klasse einer bestimmten Konfiguration löschen (**wlm_delete_class**)

Die Änderungen werden nur auf Eigenschaftendateien der angegebenen WLM-Konfiguration angewendet. Durch Angabe einer leeren Zeichenfolge für den Konfigurationsnamen ist es möglich, die Änderung nur auf die Klassen im Speicher anzuwenden, um eine sofortige Aktualisierung des aktiven Konfigurationsstatus zu erreichen.

Für den Aufruf der API-Aufrufe muss der Aufrufende dieselben Berechtigungen wie für die Befehlszeile oder die Schnittstelle SMIT besitzen. Dies sind im Einzelnen:

- Jeder Benutzer kann die Klassennamen und -merkmale lesen.
- Nur Root kann Superklassen erstellen, ändern und löschen.
- Nur Root oder designierte Superklassenadministratoren (Superklassenattribute **adminuser** und **admingroup**) können Unterklassen einer bestimmten Superklasse erstellen, ändern und löschen.

In den Fällen, in denen die WLM-Verwaltung über die Befehlszeile und Verwaltungstools von WLM-Administratoren und über die API von Anwendungen durchgeführt wird, ist Vorsicht geboten. Beide Schnittstellen verwenden denselben Namespace für die Superklassen- und Unterklassennamen und dieselbe Gesamtanzahl an Superklassen und Unterklassen.

Wenn die API die WLM-Daten im Speicher direkt ändert (z. B. neue Klassen erstellt), ist außerdem zu beachten, dass die WLM-Administratoren erst dann Kenntnis von dieser Änderung erhalten, wenn Klassen, die sie selbst nicht erstellt haben, in der Ausgabe von Befehlen wie **wlmstat** erscheinen. Um zu vermeiden, dass Anwendungen, die diese API verwenden, in Konflikte gestürzt werden, wenn der Systemadministrator WLM aktualisiert, werden die Klassen, die über die API erstellt werden und nicht in den WLM-Eigenschaftendateien definiert sind, nicht automatisch aus dem Speicher entfernt. Sie bleiben so lange aktiv, bis sie explizit mit der Routine **wlm_delete_class** oder über einen Aufruf des Befehls **rmclass** (direkter Aufruf oder Aufruf über SMIT durch den Systemadministrator) entfernt werden.

Außerdem ermöglichen die Anwendungsprogrammiersstellen (API, Application Programming Interface) von WLM Anwendungen Folgendes:

- Betriebsart von WLM mit der Funktion **wlm_set** abfragen oder ändern
- Aktuellen Status von WLM abfragen
- WLM stoppen
- Zwischen aktivem und passivem Modus hin- und herwechseln
- **rset**-Bindung aktivieren und inaktivieren
- WLM mit der Routine **wlm_load** unter Verwendung der aktuellen oder einer alternativen Konfiguration starten oder aktualisieren
- Einen Prozess oder eine Prozessgruppe mit der Routine **wlm_assign** einer Klasse zuordnen

Die APIs setzen dieselben Berechtigungen voraus wie die entsprechenden Befehle **wlmcntrl** und **wlmassign**:

- Jeder Benutzer kann den Status von WLM abfragen.
- Nur Root kann die Betriebsart von WLM ändern.
- Nur Root kann eine vollständige Konfiguration aktualisieren.
- Root und berechtigte Superklassenadministratoren (**adminuser** und **admingroup**) können WLM für die Unterklassen einer bestimmten Superklasse aktualisieren.
- Root, berechtigte Benutzer (angegeben mit **authuser** oder **authgroup**) und berechtigte Superklassenadministratoren (**adminuser** und **admingroup**) können Prozesse einer Superklasse oder Unterklasse zuordnen.

WLM-API für Statistiken

Die Routinen der WLM-API und **wlm_get_bio_stats** ermöglichen Anwendungen den Zugriff auf die WLM-Statistiken, die von den Befehlen **wlmstat** angezeigt werden.

WLM-API für Klassifizierung

Mit der Routine **wlm_check** kann der Benutzer die Klassendefinitionen und Zuordnungsregeln für eine bestimmte WLM-Konfiguration prüfen. Eine Anwendung kann mit der API-Routine **wlm_classify** feststellen, welcher Klasse ein Prozess mit einer bestimmte Attributgruppe zugeordnet werden soll.

Zugehörige Informationen:

wlmassign command

Binärkompatibilität:

Zur Unterstützung der Binärkompatibilität für den Fall, dass künftig Änderungen an den Datenstrukturen vorgenommen werden, wird an jeden API-Aufruf eine Versionsnummer als Parameter übergeben.

Damit ist die Bibliothek in der Lage festzustellen, mit welcher Version der Datenstrukturen die Anwendung erstellt worden ist.

Beispiele für die Klassifizierung, Regeln und Grenzwerte in Workload Manager

Es gibt verschiedene Methoden für die Klassifizierung eines Prozesses, und alle können gleichzeitig angewendet werden.

Für eine maximale Konfigurationsflexibilität wird ein Topdown-Algorithmus mit strikter Erstübereinstimmung verwendet. Sie können Prozessgruppen nach Benutzer mit Fallunterscheidungen für Programme mit bestimmten Namen, nach Pfad mit Fallunterscheidungen für bestimmte Benutzer oder nach beliebigen anderen Schemata bilden.

Beispiel für Zuordnungsregeln in Workload Manager:

Dieses Beispiel zeigt eine Datei *rules* der höchsten Ebene für die Konfiguration *Config* (Datei */etc/wlm/Config/rules*).

```
* Diese Datei enthält die Regeln, die WLM verwendet,
* um einen Prozess einer Superklasse zuzuordnen.
*
* class resvd user    group  application    type  tag
db1      -    -      -      /usr/bin/oracle*  _DB1
db2      -    -      -      /usr/bin/oracle*  _DB2
devlt    -    -      dev      -          -    -
VPS      -    bob,ted  -          -          -    -
acctg    -    -      acct*    -          -    -
System   -    root     -          -          -    -
Default  -    -      -          -          -    -
```

Im Folgenden sehen Sie ein Beispiel für die Datei *rules* für die Superklasse **devlt** in der Datei */etc/wlm/Config/devlt/rules*:

```
* Diese Datei enthält die Regeln, die WLM verwendet, um
* einen Prozess zu einer Unterklasse der Superklasse
* devlt zuzuordnen.
*
* class resvd user    group  application          type      tag
hackers  -    jim,liz  -          -              -          -
hogs     -    -        -          -              64bit+plock -
editors  -    !sue    -          /bin/vi,/bin/emacs -          -
build    -    -        -          /bin/make,/bin/cc  -          -
Default  -    -        -          -              -          -
```

Anmerkung: Der Stern (*) ist das Kommentarzeichen, das in der Datei *rules* verwendet wird.

Im Folgenden finden Sie Beispiele, die diese Datei *rules* verwenden. Die folgenden Beispiele gehen davon aus, dass das Vererbungsattribut (inheritance) für die beschriebenen Superklassen und Unterklassen nicht aktiviert ist. Wenn die Vererbung aktiviert ist, erben neue Prozesse die Superklasse oder Unterklasse von ihren übergeordneten Prozessen.

- Wenn der Benutzer *joe* aus der Gruppe *acct3* den Befehl **/bin/vi** ausführt, wird der Prozess der Superklasse *acctg* zugeordnet.

- Wenn der Benutzer sue aus der Gruppe dev den Befehl `/bin/emacs` ausführt, wird der Prozess der Superklasse devlt (Übereinstimmung der Gruppen-ID), aber nicht der Unterklasse editors zugeordnet, weil der Benutzer von dieser Klasse ausgeschlossen ist. Der Prozess wird standardmäßig devlt zugeordnet.
- Wenn ein Datenbankadministrator mit der Benutzer-ID oracle und der Gruppen-ID dbm den Befehl `/usr/sbin/oracle` startet, um die Datenbank DB1 bereitzustellen, wird der Prozess der Superklasse Default zugeordnet. Nur wenn der Prozess seine Kennung auf `_DB1` setzt, wird er der Superklasse db1 zugeordnet.

Beispiel für Klassen von Workload Manager mit Anteilen und Grenzwerten:

Nehmen wir für dieses Beispiel an, dass die Klassen A, B, C und D jeweils Anteile von 3, 2, 1 und 1 haben.

Wenn die Klassen A, C und D aktiv sind, ergibt die Berechnung der Ziele Folgendes:

$$\text{Ziel(A)} = 3/5 = 60 \%$$

$$\text{Ziel(C)} = 1/5 = 20 \%$$

$$\text{Ziel(D)} = 1/5 = 20 \%$$

Wenn während der Tests festgestellt wird, dass die Anwendungen in der Klasse A bei einer zulässigen Ressourcennutzung von 50 % eine angemessene Leistung erbringen, empfiehlt es sich, die anderen 50 % der Ressource anderen Klassen zur Verfügung zu stellen. Sie können dies erreichen, indem Sie für die Klasse A einen variablen Maximalwert von 50 % für diese Ressource festlegen. Da das derzeit berechnete Ziel von 60 % über diesem Grenzwert liegt, wird er nach unten an den variablen Maximalwert angeglichen. In diesem Fall wird das Ziel bzw. der tatsächliche Verbrauch (je nachdem, welcher Wert kleiner ist) der Klasse A von der verfügbaren Ressourcenmenge abgezogen. Da diese Klasse jetzt ein Ziel hat, das durch einen Grenzwert (und nicht durch seine Anteile) beschränkt ist, werden auch die Anteile für die Klasse von der Anzahl aktiver Anteile abgezogen. Falls die Klasse A beispielsweise einen aktuellen Verbrauch von 48 % hat, ergeben sich dadurch die folgenden Ziele:

$$\text{Ziel(A)} = 3/5 = 60 \%, \text{ veränderlicher Maximalwert} = 50, = 50 \%$$

$$\text{Ziel(C)} = 1/2 * (100 - 48) = 26 \%$$

$$\text{Ziel(D)} = 1/2 * (100 - 48) = 26 \%$$

Zu einem späteren Zeitpunkt können alle Klassen aktiv werden, woraufhin die Ziele automatisch erneut angepasst werden:

$$\text{Ziel(A)} = 3/7 = 42 \%$$

$$\text{Ziel(B)} = 2/7 = 28 \%$$

$$\text{Ziel(C)} = 1/7 = 14 \%$$

$$\text{Ziel(D)} = 1/7 = 14 \%$$

Beispiel für Klassen von Workload Manager mit CPU-Grenzwerten:

In diesem Beispiel wird die CPU-Zuordnung unter der Voraussetzung untersucht, dass jede Klasse die gesamten bereitgestellten CPU-Ressourcen in Anspruch nehmen kann.

Die beiden Klassen A und B befinden sich in derselben Schicht. Die CPU-Grenzwerte für A sind [30 % - 100 %]. Die CPU-Grenzwerte für B sind [20 % - 100 %]. Wenn beide Klassen aktiv sind und genügend CPU verbrauchen, stellt WLM zuerst sicher, dass beide Klassen jede Sekunde ihre Mindestprozentsätze erhalten (durchschnittlich über mehrere Sekunden hinweg). Anschließend verteilt WLM die verbleibenden CPU-Zyklen auf der Basis der festgelegten Werte für die CPU-Zielanteile.

Wenn die CPU-Zielanteile für A und B 60 % bzw. 40 % sind, stabilisiert sich die CPU-Auslastung für A und B bei 60 % bzw. 40 %.

Es wird eine dritte Klasse, C, hinzugefügt. Diese Klasse ist eine Gruppe von CPU-lastigen Jobs und muss mit ungefähr der Hälfte (oder mehr) der verfügbaren CPU ausgeführt werden. Für Klasse C sind die Grenzwerte [20 % - 100 %] und CPU-Zielanteile von 100 % festgelegt. Wenn sich C in derselben Schicht wie A und B befindet, sehen A und B ihre CPU-Zuordnung rapide sinken, und die drei Klassen stabilisieren sich bei 30 %, 20 % bzw. 50 %. Die Anteile für A und B sind in diesem Fall gleichzeitig auch ihre Mindestwerte.

Ein Systemadministrator möchte unter Umständen jedoch nicht, dass Stapeljobs 50 % der CPU verbrauchen, wenn gleichzeitig Jobs, möglicherweise mit einer höheren Priorität ausgeführt werden. In einer Situation wie der im vorherigen Beispiel dargestellten wird Klasse C in eine Schicht mit einer niedrigeren Priorität verschoben. Klasse C erhält dann so viel CPU, wie übrig bleibt, wenn die Anforderungen von A und B erfüllt worden sind. Im obigen Beispiel erhält C keine CPU-Zeit, weil A und B jeweils 100 % der CPU verbrauchen können. In den meisten Situationen enthalten A und B, die sich in einer Schicht mit einer höheren Priorität befinden, jedoch interaktive oder transaktionsorientierte Jobs, die nicht ständig die gesamte CPU beanspruchen. Damit erhält die Klasse C einen gewissen Anteil der CPU, um den sie mit anderen Klassen in derselben oder in niedrigeren Schichten konkurrieren muss.

Beispiel für Klassen von Workload Manager mit Speicherbegrenzungen:

Dieses Beispiel untersucht die Speicherzuordnung zu Prozessgruppen mit variierenden Speicherzielen.

Es müssen drei Prozessgruppen ausgeführt werden: eine Gruppe interaktiver Prozesse, die ausgeführt werden müssen, wenn sie verwendet werden (PEOPLE), ein Stapeljob, der immer im Hintergrund ausgeführt wird (BATCH1), und ein zweiter, wichtigerer Stapeljob, der jede Nacht ausgeführt wird (BATCH0).

PEOPLE hat einen angegebenen Speichermindestwert von 20 %, ein Speicherziel von 50 Anteilen und einen Klassenschichtwert von 1. Der Mindestwert von 20 % stellt sicher, dass die Desktopanwendungen in dieser Klasse relativ schnell reagieren, wenn Benutzer Tastatureingaben vornehmen.

BATCH1 hat einen Speichermindestwert von 50 %, ein Speicherziel von 50 Anteilen und einen Schichtwert von 3.

BATCH0 hat einen Speichermindestwert von 80 %, ein Speicherziel von 50 Anteilen und einen Schichtwert von 2.

Die Klassen PEOPLE und BATCH1 haben einen Gesamtspeichermindestwert von 70. Im normalen Betrieb (wenn BATCH0 nicht aktiv ist) können beide Klasse den gesamten für sie reservierten Speicher erhalten. Sie teilen sich den Rest des Speichers in der Maschine jeweils zur Hälfte, obwohl sie sich in unterschiedlichen Schichten befinden. Wenn BATCH0 um Mitternacht gestartet wird, erreicht der Gesamtspeichermindestwert 150. WLM ignoriert die Mindestanforderungen für die untersten Schichten so lange, bis Prozesse in den oberen Schichten beendet werden. BATCH0 erhält Speicher aus der 50%-Reserve von BATCH1, aber nicht aus der 20%-Reserve von PEOPLE. Nach Abschluss von BATCH0 werden die Speicherreserven für die Prozesse der Schicht 3 wieder berücksichtigt, und das System kehrt zu seiner normalen gleichmäßigen Speicherverteilung zurück.

Workload-Manager-Befehle

WLM stellt Befehle bereit, mit denen Systemadministratoren eine Vielzahl von Funktionen ausführen können.

Beispiele für solche Funktionen sind:

- Superklassen und Unterklassen mit den Befehlen **mkclass**, **chclass** und **rmclass** erstellen, ändern und löschen. Diese Befehle aktualisieren die Dateien *classes*, *shares* und *limits*.
- WLM mit dem Befehl **wlmcntrl** starten, stoppen und aktualisieren.

- WLM-Eigenschaftendateien mit dem Befehl **wlmcheck** für eine bestimmte Konfiguration überprüfen und feststellen, welcher Klasse (Superklasse und Unterklasse) ein Prozess mit einer bestimmten Gruppe von Attributen zugeordnet ist.
- Ressourcenauslastung einer Klasse mit dem Befehl **wlmstat** (ASCII) überwachen. Die meisten Leistungstools, z. B. die Tools, die mit den Befehlen **svmon** und **topas** gestartet werden, haben Erweiterungen, die die WLM-Klassen berücksichtigen und unter Verwendung neuer Befehlszeilenoptionen Statistiken pro Klasse und pro Schicht bereitstellen.
- Die Flags des Befehls **ps** ermöglichen dem Benutzer anzuzeigen, zu welcher Klasse ein Prozess und seine Anwendungskennung gehören. Mit dem Befehl **ps** kann der Benutzer außerdem alle Prozesse auflisten, die zu einer bestimmten Superklasse oder Unterklasse gehören.
- Für die Verwaltung der Zuordnungsregeln ist keine Befehlszeilenschnittstelle verfügbar. Sie müssen das Verwaltungstool SMIT oder einen Texteditor verwenden.

Einheitenknoten

Einheiten sind in Clustern, so genannten *Knoten*, organisiert. Jeder Knoten ist ein logisches Subsystem mit Einheiten, in dem die Einheiten der unteren Ebenen von den Einheiten der höheren Ebenen abhängig sind. Es bestehen also Beziehungen zwischen untergeordneten und übergeordneten Einheiten.

Der Systemknoten ist beispielsweise der Knoten der höchsten Ebene und umfasst alle physischen Einheiten im System. Die Systemeinheit bildet den Ausgangspunkt des Knotens. Darunter befinden sich der Bus und die Adapter, die von der Systemeinheit der höheren Ebene abhängig sind. Unten in der Hierarchie befinden sich alle Einheiten, an die keine anderen Einheiten angeschlossen sind. Diese Einheiten sind von allen Einheiten auf höheren Ebenen der Hierarchie abhängig.

Während des Systemstarts werden die Abhängigkeiten zwischen übergeordneten und untergeordneten Einheiten verwendet, um alle Einheiten zu konfigurieren, aus denen sich ein Knoten zusammensetzt. Die Knoten werden ausgehend vom Knoten der Ausgangsebene konfiguriert. Alle Einheiten, die von einer Einheit einer höheren Ebene abhängig sind, werden erst konfiguriert, wenn die Konfiguration der Einheit der höheren Ebene abgeschlossen ist.

Das Betriebssystem AIX unterstützt das Feature MPIO (Multiple Path I/O). Wenn eine Einheit einen MPIO-fähigen Einheitentreiber hat, kann sie mehrere übergeordnete Einheiten in dieser Hierarchie haben. Damit sind mehrere gleichzeitige Kommunikationspfade zwischen der Einheit und einer bestimmten Maschine oder logischen Partition auf einer Maschine möglich.

Einheitenklassen

Für die Verwaltung von Einheiten muss das Betriebssystem verstehen, welche Einheitenverbindungen zulässig sind. Das Betriebssystem klassifiziert Einheiten hierarchisch in drei Gruppen.

Die drei Gruppen sind im Folgenden aufgelistet:

- Funktionale Klassen
- Funktionale Unterklassen
- Einheitentypen

Funktionale Klassen setzen sich aus Einheiten zusammen, die dieselbe Funktion ausführen. Eine funktionale Klasse sind beispielsweise Drucker. Funktionale Klassen werden nach bestimmten Einheitenähnlichkeiten in Unterklassen unterteilt. Drucker haben beispielsweise eine serielle oder eine parallele Schnittstelle. Serielle Drucker bilden eine Unterklasse und parallele Drucker eine andere. Einheitentypen werden nach Modell und Hersteller klassifiziert.

Einheitenklassen definieren gültige Verbindungen zwischen übergeordneten und untergeordneten Einheiten für das Betriebssystem. Die Hierarchie definiert die möglichen Unterklassen, die an jeder der möglichen

untergeordneten Verbindungspositionen zulässig sind. Der Begriff "RS-232 8-Port Adapter" gibt beispielsweise an, dass nur Einheiten, die zur Unterklasse RS-232 gehören, an jeden der acht Anschlüsse des Adapters angeschlossen werden können.

Einheitenklassen und ihre hierarchischen Abhängigkeiten werden in einer ODM-Einheitenkonfigurationsdatenbank (Object Data Manager) verwaltet.

Einheitenkonfigurationsdatenbank und Einheitenverwaltung

Einheitendaten sind in einer vordefinierten oder angepassten Datenbank enthalten, die als Einheitenkonfigurationsdatenbank verwendet wird.

Die vordefinierte Datenbank enthält Konfigurationsdaten für alle möglichen Einheiten, die vom System unterstützt werden. Die hierarchischen Einheitenklassendaten sind in dieser Datenbank enthalten. Die angepasste Datenbank enthält Konfigurationsdaten für alle derzeit definierten und konfigurierten Einheiten im System. Für jede derzeit an das System angeschlossene Einheit wird ein Datensatz geführt.

Der Konfigurationsmanager ist ein Programm, das Einheiten in Ihrem System automatisch während des Systemstarts und zur Laufzeit konfiguriert. Der Konfigurationsmanager verwendet während dieses Prozesses die Daten aus der vordefinierten und der angepassten Datenbank und aktualisiert anschließend die angepasste Datenbank.

Die Funktion MPIO (Multiple-Path I/O) verwendet eine UDID (Unique Device Identifier, Eindeutige Einheiten-ID), um alle MPIO-fähigen Einheiten, unabhängig vom Pfad, in dem sie gefunden wurden, zu identifizieren. Die UDID wird in der Einheitenkonfigurationsdatenbank gespeichert. Wenn eine Einheit erkannt wird, werden die UDIDs in der Datenbank geprüft, um festzustellen, ob die Einheit neu ist oder ob die erkannte Einheit nur ein anderer Pfad zu einer vorhandenen Einheit ist. Wenn mehrere Pfade zu einer Einheit erkannt werden, entscheidet der Einheitentreiber oder die Kernelerweiterung Path Control Manager (Pfadsteuerungsmanager), welcher Pfad für eine bestimmte Anforderung verwendet wird.

Sie können SMIT oder Betriebssystembefehle verwenden, um Einheitenverwaltungstasks wie das Löschen, Hinzufügen oder Konfigurieren einer Einheit auszuführen.

Zugehörige Konzepte:

„Verwaltung MPIO-fähiger Einheiten“ auf Seite 577

Die Funktion Multiple Path I/O (MPIO) kann verwendet werden, um für Failover-Zwecke alternative Pfade zu einer Einheit zu definieren.

Zugehörige Tasks:

„Installation einer Einheit vorbereiten“ auf Seite 410

Die Installation von Einheiten im System setzt sich aus den folgenden Schritten zusammen: Anschlussposition für die Einheit ermitteln, Einheit physisch anschließen, Einheit mit dem Konfigurationsmanager oder SMIT konfigurieren.

Einheitenstatus

Einheiten, die mit dem System verbunden sind, können vier Status annehmen.

Einheiten, die mit dem System verbunden sind, können einen der folgenden Status annehmen:

| Eintrag | Beschreibung |
|-----------------|---|
| Nicht definiert | Die Einheit ist dem System nicht bekannt. |
| Definiert | Es sind bestimmte Informationen über die Einheit in der angepassten Datenbank protokolliert, aber die Einheit ist für das System nicht verfügbar. |
| Verfügbar | Eine definierte Einheit ist mit dem Betriebssystem verbunden, oder die Einheit ist konfiguriert. |
| Gestoppt | Die Einheit ist nicht verfügbar, aber über ihren Einheitentreiber bekannt. |

Wenn ein nicht grafikfähiges Terminal und ein Drucker abwechselnd denselben Anschluss am nicht grafikfähigen Terminal verwenden, sind ein nicht grafikfähiges Terminal und ein Drucker für dieselbe über-

geordnete Einheit und denselben Anschluss in der Einheitenkonfigurationsdatenbank definiert. Es kann jeweils nur eine dieser Einheiten konfiguriert sein. Wenn der Anschluss des nicht grafikfähigen Terminals konfiguriert wird, werden die druckerspezifischen Konfigurationsdaten so lange aufbewahrt, bis dieser erneut konfiguriert wird. Die Einheit wird nicht entfernt, sondern nimmt den Status "definiert" an. Wenn eine Einheit im Status "definiert" verwaltet wird, bleiben die angepassten Informationen für eine Einheit, die derzeit nicht im Gebrauch ist (bevor sie erstmalig verfügbar gemacht wird oder während sie vorübergehend aus dem System entfernt wird), erhalten.

Wenn ein Einheitentreiber für eine Einheit vorhanden ist, kann die Einheit über den Einheitentreiber verfügbar gemacht werden.

Einige Einheiten, insbesondere TCP/IP-Pseudoeinheiten, benötigen den Status "gestoppt".

Zugehörige Tasks:

„Asynchrone Adapter dekonfigurieren“ auf Seite 594
Sie können einen asynchronen Adapter dekonfigurieren.

Positionscode für Einheiten

Der *Positionscode* ist ein Pfad, der vom CPU-Einschub bzw. der Systemeinheit über den Adapter, das Signalkabel und den Verteilerkasten für asynchrone Anschlüsse (sofern vorhanden) zur Einheit bzw. Workstation führt. Dieser Code ist eine weitere Möglichkeit für die Identifizierung physischer Einheiten.

Der Positionscode setzt sich je nach Einheitentyp aus bis zu vier Feldern mit Informationen zusammen. Diese Felder geben den Einschub, den Steckplatz, den Anschluss und die Position an. Jedes dieser Felder enthält einen zweistelligen Wert.

Der Positionscode eines Einschubs enthält lediglich das Feld für den Einschub und ist ein einfacher zweistelliger Code. Der Positionscode eines Adapters setzt sich aus den Feldern für den Einschub und den Steckplatz zusammen und hat das Format *AA-BB*, wobei *AA* der Einschubposition entspricht und *BB* den Bus und den Steckplatz mit dem Adapter angibt. Andere Einheiten haben Positionscode mit dem Format *AA-BB-CC* oder *AA-BB-CC-DD*. *AA-BB* gibt den Positionscode des Adapters an, an den die Einheit angeschlossen ist, *CC* entspricht dem Anschluss am Adapter, mit dem die Einheit verbunden ist, und *DD* gibt eine Position oder SCSI-Einheitenadresse an.

Zugehörige Informationen:

Teilepositionen und Positionscode

Positionscode für Adapter

Der Positionscode für einen Adapter setzt sich aus zwei Ziffernpaaren im Format *AA-BB* zusammen. *AA* gibt den Positionscode des Einschubs mit dem Adapter an und *BB* den E/A-Bus und den Steckplatz mit der Karte.

Wenn das Feld **AA** den Wert 00 hat, befindet sich der Adapter je nach Systemtyp im CPU-Einschub oder in der Systemeinheit. Jeder andere Wert im Feld **AA** gibt an, dass sich die Karte in einem Einschub für E/A-Erweiterungen befindet. In diesem Fall gibt der Wert des Feldes **AA** den E/A-Bus und die Nummer des Steckplatzes im CPU-Einschub an, der den asynchronen Erweiterungsadapter enthält. Die erste Ziffer steht für den E/A-Bus, wobei 0 dem Standard-E/A-Bus und 1 dem optionalen E/A-Bus entspricht. Die zweite Ziffer gibt die Nummer des Steckplatzes im angegebenen E/A-Bus an.

Die erste Ziffer des Feldes **BB** gibt die E/A-Platine an, die die Adapterkarte enthält. Wenn sich die Karte im CPU-Einschub oder in der Systemeinheit befindet, entspricht diese Ziffer dem Wert 0 für den Standard-E/A-Bus bzw. dem Wert 1 für den optionalen E/A-Bus. Befindet sich die Karte in einem Einschub für E/A-Erweiterungen, ist die erste Ziffer eine 0. Die zweite Ziffer gibt die Nummer des Steckplatzes im angegebenen E/A-Bus (bzw. die Nummer des Steckplatzes im Einschub für E/A-Erweiterungen) an, der die Karte enthält.

Der Positionscode 00-00 steht für die Identifikation der Standard-E/A-Platine.

Beispiele:

| Eintrag | Beschreibung |
|---------|--|
| 00-05 | Gibt eine Adapterkarte in Steckplatz 5 der Standard-E/A-Platine an. Der Adapter befindet sich je nach Systemtyp entweder im CPU-Einschub oder in der Systemeinheit. |
| 00-12 | Gibt einen Adapter in Steckplatz 2 des optionalen E/A-Busses im CPU-Einschub an. |
| 18-05 | Gibt eine Adapterkarte in Steckplatz 5 eines Einschubs für E/A-Erweiterungen an. Der Einschub ist mit dem asynchronen Erweiterungsadapter in Steckplatz 8 des optionalen E/A-Busses im CPU-Einschub verbunden. |

Zugehörige Informationen:

Teilepositionen und Positionscodes

Positionscodes für Drucker und Plotter

Der Positionscodes 00-00-S1-00 bzw. 00-00-S2-00 gibt an, dass der Drucker, der Plotter oder das nicht grafikfähige Terminal an den seriellen Anschluss s1 bzw. s2 auf der Standard-E/A-Platine angeschlossen ist. Der Positionscodes 00-00-0P-00 gibt an, dass der Paralleldrucker an den Parallelanschluss auf der Standard-E/A-Platine angeschlossen ist.

Jeder andere Positionscodes gibt an, dass der Drucker, der Plotter bzw. das nicht grafikfähige Terminal an eine andere Adapterkarte als die Standard-E/A-Platine angeschlossen ist. Für diese Drucker, Plotter und nicht grafikfähigen Terminals hat der Positionscodes das Format AA-BB-CC-DD. AA-BB gibt den Positionscodes des steuernden Adapters an.

| Eintrag | Beschreibung |
|---------|---|
| AA | Der Wert 00 im Feld AA zeigt an, dass sich die Adapterkarte je nach Systemtyp im CPU-Einschub oder in der Systemeinheit befindet. Jeder andere Wert im Feld AA gibt an, dass sich die Karte in einem Einschub für E/A-Erweiterungen befindet. In diesem Fall gibt die erste Stelle den E/A-Bus und die zweite Stelle die Nummer des Steckplatzes im Bus des CPU-Einschubs an, der den asynchronen Erweiterungsadapter enthält, an den der E/A-Erweiterungseinschub angeschlossen ist. |
| BB | Die erste Ziffer des Feldes BB gibt den E/A-Bus an, der die Adapterkarte enthält. Wenn sich die Karte im CPU-Einschub oder in der Systemeinheit befindet, entspricht diese Ziffer dem Wert 0 für den Standard-E/A-Bus bzw. dem Wert 1 für den optionalen E/A-Bus. Befindet sich die Karte in einem Einschub für E/A-Erweiterungen, ist die erste Ziffer eine 0. Die zweite Ziffer gibt die Nummer des Steckplatzes im E/A-Bus (bzw. die Nummer des Steckplatzes im Einschub für E/A-Erweiterungen) an, der die Karte enthält. |
| CC | Das Feld CC gibt den Anschluss an der Adapterkarte an, an den der Verteilerkasten für asynchrone Anschlüsse angeschlossen ist. Die gültigen Werte sind 01, 02, 03 und 04. |
| DD | Das Feld DD gibt die Nummer der Position am Verteilerkasten für asynchrone Anschlüsse an, an die der Drucker, der Plotter bzw. das nicht grafikfähige Terminal angeschlossen ist. |

Zugehörige Informationen:

Teilepositionen und Positionscodes

Positionscodes für nicht grafikfähige Terminals

Der Positionscodes 00-00-S1-00 bzw. 00-00-S2-00 gibt an, dass das nicht grafikfähige Terminal an den seriellen E/A-Standardanschluss s1 bzw. s2 angeschlossen ist.

Jeder andere Positionscodes gibt an, dass das nicht grafikfähige Terminal an eine andere Adapterkarte als die Standard-E/A-Platine angeschlossen ist. Für diese Einheiten hat der Positionscodes das Format AA-BB-CC-DD. AA-BB gibt den Positionscodes der steuernden Adapterkarte an.

| Eintrag | Beschreibung |
|----------------|--|
| AA | Der Wert 00 im Feld AA zeigt an, dass sich die Adapterkarte je nach Systemtyp im CPU-Einschub oder in der Systemeinheit befindet. Jeder andere Wert im Feld AA gibt an, dass sich die Karte in einem Einschub für E/A-Erweiterungen befindet. In diesem Fall gibt die erste Stelle den E/A-Bus und die zweite Stelle die Nummer des Steckplatzes im Bus des CPU-Einschubs an, der den asynchronen Erweiterungsadapter enthält, an den der E/A-Erweiterungseinschub angeschlossen ist. |
| BB | Die erste Ziffer des Feldes BB gibt den E/A-Bus an, der die Adapterkarte enthält. Wenn sich die Karte im CPU-Einschub oder in der Systemeinheit befindet, entspricht diese Ziffer dem Wert 0 für den Standard-E/A-Bus bzw. dem Wert 1 für den optionalen E/A-Bus. Befindet sich die Karte in einem Einschub für E/A-Erweiterungen, ist die erste Ziffer eine 0. Die zweite Ziffer gibt die Nummer des Steckplatzes im E/A-Bus (bzw. die Nummer des Steckplatzes im Einschub für E/A-Erweiterungen) an, der die Karte enthält. |
| CC | Das Feld CC gibt den Anschluss an der Adapterkarte an, an den der Verteilerkasten für asynchrone Anschlüsse angeschlossen ist. Die gültigen Werte sind 01, 02, 03 und 04. |
| DD | Das Feld DD gibt die Nummer der Position am Verteilerkasten für asynchrone Anschlüsse an, an die das nicht grafikfähige Terminal angeschlossen ist. |

Zugehörige Informationen:

Teilepositionen und Positionscode

Positionscode für SCSI-Einheiten

Im Folgenden sind die Positionscode für SCSI-Einheiten beschrieben.

Diese Positionscode gelten für alle SCSI-Einheiten, einschließlich der folgenden:

- CD-ROMs
- Platten
- Initiatoreinheiten
- Laufwerke für wieder beschreibbare optische Platten
- Bänder
- Zielmodus

Der Positionscode hat das Format AA-BB-CC-S,L. **AA-BB** gibt den Positionscode des SCSI-Adapters an, der die SCSI-Einheit steuert.

| Eintrag | Beschreibung |
|----------------|--|
| AA | Der Wert 00 im Feld AA zeigt an, dass sich die steuernde Adapterkarte je nach Systemtyp im CPU-Einschub oder in der Systemeinheit befindet. |
| BB | Das Feld BB gibt den E/A-Bus und den Steckplatz mit der Karte an. Die erste Ziffer gibt den E/A-Bus an. 0 steht für den Standard-E/A-Bus und 1 für den optionalen E/A-Bus. Die zweite Ziffer ist der Steckplatz im angegebenen E/A-Bus, der die Karte enthält. Der Wert 00 im Feld BB steht für den Standard-SCSI-Controller. |
| CC | Das Feld CC gibt den SCSI-Bus der Karte an, an die die Einheit angeschlossen ist. Wenn eine Karte nur einen einzigen SCSI-Bus hat, ist der Wert für dieses Feld 00. Ansonsten gibt der Wert 00 an, dass eine Einheit an den internen SCSI-Bus der Karte angeschlossen ist. Der Wert 01 gibt an, dass eine Einheit an den externen SCSI-Bus der Karte angeschlossen ist. |
| S,L | Das Feld S,L gibt die SCSI-ID und die LUN der SCSI-Einheit an. Der Wert S steht für die SCSI-ID und L für die LUN. |

Zugehörige Informationen:

Teilepositionen und Positionscode

Positionscode für Wertgeber und beleuchtete Funktionstasten

Für eine an einen Grafikeingabeadapter angeschlossene Einheit mit Wertgebern oder beleuchteten Funktionstasten ist das Format des Positionscode wie folgt: AA-BB-CC.

Die einzelnen Felder werden wie folgt interpretiert:

| Eintrag | Beschreibung |
|---------|--|
| AA | Der Wert 00 im Feld AA zeigt an, dass sich die steuernde Adapterkarte je nach Systemtyp im CPU-Einschub oder in der Systemeinheit befindet. |
| BB | Das Feld BB gibt den E/A-Bus und den Steckplatz mit der Karte an. Die erste Ziffer gibt den E/A-Bus an. 0 steht für den Standard-E/A-Bus und 1 für den optionalen E/A-Bus. Die zweite Ziffer ist der Steckplatz im angegebenen E/A-Bus, der die Karte enthält. |
| CC | Das Feld CC gibt die Kartensteckverbindung an, an die die Einheit angeschlossen ist. Die gültigen Werte sind 01 und 02, je nachdem, ob die angeschlossene Einheit sich am Anschluss 1 oder am Anschluss 2 auf der Karte befindet. |

Anmerkung: Für seriell angeschlossene Einheiten mit Wertgebern oder beleuchteten Funktionstasten wird kein Positionscode angegeben. Bei diesen Einheiten wird angenommen, dass sie an ein nicht grafikfähiges Terminal angeschlossen sind. Das nicht grafikfähige Terminal wird vom Benutzer während der Definition der Wertgeber und beleuchteten Funktionstasten angegeben.

Zugehörige Informationen:

Teilepositionen und Positionscode

Positionscode für Multiprotokollanschluss

Der Positionscode für einen Multiprotokollanschluss hat das Format *AA-BB-CC-DD*. *AA-BB* gibt den Positionscode für die Multiprotokolladapterkarte an.

Die einzelnen Felder werden wie folgt interpretiert:

| Eintrag | Beschreibung |
|---------|--|
| AA | Der Wert 00 im Feld AA zeigt an, dass sich die Multiprotokolladapterkarte je nach Systemtyp im CPU-Einschub oder in der Systemeinheit befindet. |
| BB | Das Feld BB gibt den E/A-Bus und den Steckplatz mit der Karte an. Die erste Ziffer gibt den E/A-Bus an. 0 steht für den Standard-E/A-Bus und 1 für den optionalen E/A-Bus. Die zweite Ziffer ist der Steckplatz im angegebenen E/A-Bus, der die Karte enthält. |
| CC | Das Feld CC gibt den Anschluss an der Adapterkarte an, an den die Multiprotokollverteilereinheit angeschlossen ist. Der Wert ist immer 01. |
| DD | Das Feld DD gibt die Nummer des physischen Anschlusses an der Multiprotokollverteilereinheit an. Die gültigen Werte sind 00, 01, 02 und 03. |

Zugehörige Informationen:

Teilepositionen und Positionscode

iSCSI konfigurieren

Die Konfiguration von iSCSI umfasst die Konfiguration des Adapters und das Hinzufügen oder Aktualisieren von Zielen.

iSCSI-Adapter in AIX konfigurieren

Die Konfiguration des iSCSI-Adapters ist eine sehr einfache und überschaubare Aufgabe.

1. Geben Sie an der AIX-Eingabeaufforderung den Befehl **smit iscsi** ein: Die Anzeige "iSCSI" erscheint.
2. Wählen Sie in der Anzeige "iSCSI" die Option **iSCSI-Adapter** aus. Die Anzeige "iSCSI-Adapter" erscheint.
3. Wählen Sie in der Anzeige "iSCSI-Adapter" die Option **Merkmale eines iSCSI-Adapters ändern/anzeigen** aus. Die Anzeige "Merkmale eines iSCSI-Adapters ändern/anzeigen" erscheint.
4. Wählen Sie in der Liste den iSCSI-Adapter aus, den Sie konfigurieren möchten. Es erscheint eine Konfigurationsanzeige wie die folgende:

| | |
|---------------|-----------------|
| | [Eingabefelder] |
| iSCSI-Adapter | ics0 |
| Beschreibung | iSCSI-Adapter |

| | | | |
|---|-------------------------|---|----|
| Status | Verfügbar | | |
| Position | 10-60 | | |
| Maximale Anzahl Befehle für Adapterwarteschlange | [200] | + | +# |
| Maximale Übertragungsgröße | [0x100000] | + | |
| Dateiname für Erkennung | [/etc/iscsi/targetshw] | | |
| Erkennungs-Policy | [file] | | |
| Dateiname der Datei mit den Zugangsdaten für die automatische Erkennung | [/etc/iscsi/autosecret] | | |
| IP-Adresse des Adapters | [10.1.4.187] | | |
| Teilnetzmaske | [255.255.255.0] | | |
| Gateway-Adresse | [10.1.4.1] | | |
| Änderungen nur auf Datenbank anwenden | nein | + | |

Anmerkung: Wenn Sie eine Frage zum Zweck eines bestimmten Feldes haben, platzieren Sie den Cursor in dem Feld, und drücken Sie anschließend die Taste **F1** für Hilfe.

Wenn Sie bei der Erkennung mit unstrukturierten Dateien arbeiten möchten, geben Sie im Feld **Erkennungs-Policy** den Wert `file` ein. Zur Verwendung der Erkennung mit ODM geben Sie im Feld **Erkennungs-Policy** den Wert `odm` ein. Für iSCSI-Ziele, die mit DHCP erkannt werden, geben Sie im Feld **Erkennungs-Policy** den Wert `slp` ein.

Unstrukturierte Datei eines iSCSI-Ziels aktualisieren

Die unstrukturierte Datei ist die statische Konfigurationsdatei, die zum Konfigurieren von iSCSI-Zielen verwendet wird. Der Standarddateiname ist `/etc/iscsi/targetshw`.

In der unstrukturierten Datei müssen Sie alle relevanten Erkennungsmerkmale für iSCSI-Ziele explizit angeben.

Zugehörige Informationen:

targets File

Statisch erkannte iSCSI-Ziele zu ODM hinzufügen

Wenn die automatische Erkennung nicht verwendet wird, erhält der iSCSI-Adapter die Beschreibungen der iSCSI-Ziele aus einer unstrukturierten Datei oder aus ODM.

Sie können AIX-Befehle oder SMIT verwenden, um die Informationen für die iSCSI-Ziele in ODM zu bearbeiten. Mit den Befehlen **chiscsi**, **lsiscsi**, **mkiscsi** und **rmiscsi** können Sie Informationen zu iSCSI-Zielen in ODM ändern, anzeigen, hinzufügen bzw. entfernen.

Gehen Sie wie folgt vor, um ein statisch erkanntes iSCSI-Ziel mit SMIT zu ODM hinzuzufügen:

1. Geben Sie an der AIX-Eingabeaufforderung den Befehl **smiit iscsi** ein: Die Anzeige "iSCSI" erscheint.
2. Wählen Sie in der Anzeige "iSCSI" die Option **iSCSI Target Device Parameters in ODM** aus. Die Anzeige "iSCSI Target Device Parameters in ODM" erscheint.
3. Wählen Sie in der Anzeige "iSCSI" die Option **Add an iSCSI Target Device in ODM** aus. Die Anzeige "Add an iSCSI Target Device in ODM" erscheint.
4. Wählen Sie in der Anzeige "iSCSI" die Option **Add a iSCSI Target Device in ODM** aus. Die Anzeige "Add a iSCSI Target Device in ODM" erscheint.
5. Wählen Sie in der Liste den iSCSI-Adapter aus, den Sie konfigurieren möchten. Die Anzeige "Add a Statically Discovered iSCSI Target Device" für den ausgewählten iSCSI-Adapter erscheint.
6. Geben Sie in den Feldern die entsprechenden Informationen ein. Beispiel:

[Entry Fields]

iSCSI Adapter

ics0

| | | |
|-----------------------------|-----------------------|---|
| iSCSI Target Name | [iqn.mds9216.emc.sym] | + |
| iSCSI Group | static | + |
| IP Address of iSCSI Target | [10.1.4.25] | + |
| Port Number of iSCSI Target | [3260] | + |
| Password | [my password] | + |

Anmerkung: Wenn Sie eine Frage zum Zweck eines bestimmten Feldes haben, platzieren Sie den Cursor in dem Feld, und drücken Sie anschließend die Taste **F1** für Hilfe.

Statisch erkannte iSCSI-Ziele aus einer unstrukturierten Datei zu ODM hinzufügen

Mit SMIT können Sie die Informationen einer unstrukturierten Datei in ODM importieren.

1. Geben Sie an der AIX-Eingabeaufforderung den Befehl **smit iscsi** ein: Die Anzeige "iSCSI" erscheint.
2. Wählen Sie in der Anzeige "iSCSI" die Option **iSCSI Target Device Parameters in ODM** aus. Die Anzeige "iSCSI Target Device Parameters in ODM" erscheint.
3. Wählen Sie in der Anzeige "iSCSI" die Option **Add an iSCSI Target Device in ODM** aus. Die Anzeige "Add an iSCSI Target Device in ODM" erscheint.
4. Wählen Sie in der Anzeige "iSCSI" die Option **Add iSCSI Target Device Data in ODM from a File** aus. Die Anzeige "Add iSCSI Target Device Data in ODM from a File" erscheint.
5. Wählen Sie in der Liste den iSCSI-Adapter aus, den Sie konfigurieren möchten. Die Anzeige "Add iSCSI Target Device Data in ODM from a File" für den ausgewählten iSCSI-Adapter erscheint.
6. Geben Sie in den Feldern die entsprechenden Informationen ein. Beispiel:

| | | |
|---------------------------|------------------------|---|
| | [Entry Fields] | |
| iSCSI Protocol Device | iscsi3 | |
| iSCSI Group | [static] | + |
| Filename of iSCSI Targets | [/etc/iscsi/targetshw] | + |

Anmerkung: Wenn Sie eine Frage zum Zweck eines bestimmten Feldes haben, platzieren Sie den Cursor in dem Feld, und drücken Sie anschließend die Taste **F1** für Hilfe.

Hot-Plug-Verwaltung von PCI-Einheiten

Sie können einen neuen PCI-Hot-Plug-Adapter in einem verfügbaren PCI-Steckplatz installieren, während das Betriebssystem aktiv ist.

Es kann sich um einen weiteren Adapter desselben Typs oder einen anderen Typ von PCI-Adapter handeln. Neue Ressourcen können dem Betriebssystem und den Anwendungen bereitgestellt werden, ohne das Betriebssystem erneut starten zu müssen. Im Folgenden sind einige Gründe für das Hinzufügen eines Hot-Plug-Adapters beschrieben:

- Der vorhandenen Hardware und Firmware zusätzliche Funktionen und Kapazität hinzufügen.
- PCI-Adapter von einem System migrieren, das die von diesen Adaptern bereitgestellten Funktionen nicht mehr benötigt.
- Neues System installieren, dem die Adapterkarten zur Verfügung gestellt werden, wenn die Erstkonfiguration der optionalen Hardware subsysteme, einschließlich der PCI-Adapter, und die Installation und der Start des Betriebssystems abgeschlossen sind.

Anmerkung: Wenn Sie einen Adapter mit einer Ersetzungs- oder Hinzufügeoperation für PCI-Hot-Plug-Einheiten hinzufügen oder Dynamic Logical Partitioning verwenden, werden der Adapter und seine untergeordneten Einheiten möglicherweise nicht mit dem Befehl **bootlist** als Booteinheit aufgelistet. Unter Umständen müssen Sie die Maschine erneut starten, um dem Betriebssystem alle potenziellen Booteinheiten bekannt zu machen. Ein bereits in der Bootliste aufgeführter Adapter, der anschließend durch einen Adapter desselben Typs ersetzt wird, ist weiterhin eine gültige Booteinheit.

Sie können einen defekten oder fehlerhaften PCI-Hot-Plug-Adapter auch entfernen oder gegen einen anderen Adapter desselben Typs austauschen, ohne das System herunterzufahren oder auszuschalten. Wenn Sie den Adapter austauschen, unterstützt der vorhandene Einheitentreiber den Adapter, weil der Adapter denselben Typ hat. Die Einheitenkonfiguration und die Konfigurationsdaten zu untergeordneten Einheiten des Adapters werden für die Ersatzeinheit beibehalten. Im Folgenden sind einige Gründe für das Austauschen eines Adapters beschrieben:

- Die Karte vorübergehend austauschen, um ein Problem zu diagnostizieren oder eine fehlerhafte FRU zu isolieren.
- Einen fehlerhaften, defekten und unzuverlässigen Adapter gegen eine funktionsfähige Karte austauschen.
- Einen defekten redundanten Adapter in einer HACMP- oder MPIO-Konfiguration austauschen.

Wenn Sie einen Hot-Plug-Adapter entfernen, stehen die vom Adapter bereitgestellten Ressourcen dem Betriebssystem und den Anwendungen nicht mehr zur Verfügung. Im Folgenden sind einige Gründe für das Entfernen eines Adapters beschrieben:

- Vorhandene E/A-Subsysteme entfernen.
- Einen Adapter entfernen, der nicht mehr benötigt wird oder defekt ist, wobei keine Ersatzkarte verfügbar ist.
- Einen Adapter in ein anderes System versetzen, wenn die Funktion in dem ursprünglichen System nicht mehr benötigt wird.

Bevor Sie eine Hot-Plug-Einheit entfernen oder austauschen können, muss die Einheit dekonfiguriert werden. Der zugehörige Einheitentreiber muss alle Systemressourcen freigeben, die er der Einheit zugeordnet hat. Dazu gehören das Freigeben des Hauptspeichers, das Aufheben der Definition von Interrupt- und EPOW-Steuerroutinen, das Freigeben von DMA- und Zeitgeberressourcen sowie alle anderen erforderlichen Schritte. Der Treiber muss außerdem sicherstellen, dass Interrupts, Busspeicher und Bus-E/A in der Einheit inaktiviert werden.

Der Systemadministrator muss vor und nach dem Entfernen eines Adapters die folgenden Tasks ausführen:

- Anwendungen, Dämonprozesse oder Prozesse, die die Einheit verwenden, beenden und wiederherstellen.
- Dateisysteme abhängen und erneut anhängen.
- Einheitendefinitionen entfernen und erneut erstellen und weitere Operationen durchführen, die für die Freigabe einer verwendeten Einheit erforderlich sind.
- System in einen sicheren Zustand versetzen, in dem es gewartet werden kann.
- Alle erforderlichen Einheitentreiber anfordern und installieren.

Die Operationen für das Entfernen und Austauschen einer Einheit scheitern, wenn die Einheit, die an den angegebenen Steckplatz angeschlossen ist, nicht dekonfiguriert wurde und sich im Status "Definiert" (Defined) befindet. Hierfür können Sie den Befehl **rmdev** verwenden. Bevor Sie den Adapter in den Status "Definiert" versetzen, müssen Sie alle Anwendungen, die den Adapter verwenden, schließen, da der Befehl sonst scheitert. Weitere Informationen finden Sie in der Beschreibung des Befehls **rmdev** in der Veröffentlichung "Commands Reference".

Manchmal können Sie auch die folgenden Tasks ausführen:

- Einen PCI-Hot-Plug-Adapter vorbereiten, der installiert, entfernt oder ausgetauscht werden soll.
- Steckplätze oder PCI-Adapter ermitteln, die an der Hot-Plug-Operation beteiligt sind.
- PCI-Hot-Plug-Adapter entfernen oder installieren.

Anmerkung: Während der Hot-Plug-Operationen bleibt der Object Data Manager (ODM) gesperrt. Somit werden andere Tasks, die den ODM benötigen, blockiert oder sie schlagen fehl. Die clusterweiten Konfi-

gurationsänderungen, die von anderen Knoten eingeleitet werden, können ebenfalls blockiert werden oder fehlschlagen. Stellen Sie deshalb sicher, dass solche Tasks erst nach Abschluss der Hot-Plug-Operation ausgeführt werden.

Achtung: Obwohl die PCI-Hot-Plug-Verwaltung die Möglichkeit unterstützt, PCI-Adapter hinzuzufügen, zu entfernen und auszutauschen, ohne das System auszuschalten oder das Betriebssystem erneut zu starten, können nicht alle Einheiten in Hot-Plug-Steckplätzen auf diese Weise verwaltet werden. Beispielsweise ist es nicht möglich, die Festplatte der Datenträgergruppe rootvg oder den E/A-Controller, an den die Festplatte angeschlossen ist, zu entfernen oder auszutauschen, ohne das System auszuschalten, weil diese Platte für die Ausführung des Betriebssystems erforderlich ist. Wenn die Datenträgergruppe rootvg gespiegelt ist, können Sie den Befehl **chpv** verwenden, um die Platten offline zu schalten. Wenn sich die Datenträgergruppe rootvg auf einer oder mehreren Platten befindet, die Multi-Path I/O (MPIO) unterstützen und an mehrere E/A-Controller angeschlossen sind, kann einer dieser E/A-Controller entfernt (oder ausgetauscht) werden, ohne einen Warmstart des Systems durchführen zu müssen. In dieser Situation müssen alle vom E/A-Controller ausgehenden Pfade mit dem Befehl **rmdev -R** für den Adapter entfernt (oder ausgetauscht) werden. Dieser Befehl dekonfiguriert die Pfade und den Adapter. Anschließend können Sie mit der Hot-Plug-Verwaltung fortfahren. Bevor Sie versuchen, PCI-Hot-Plug-Adapter zu entfernen oder zu installieren, lesen Sie das Handbuch *PCI Adapter Placement Reference* (das zu den Systemeinheiten geliefert wird, die die Hot-Plug-Technologie unterstützen), um festzustellen, ob der Adapter im laufenden Betrieb ausgetauscht werden kann. Anweisungen zum Installieren und Entfernen von Adaptern finden Sie in der Dokumentation zu Ihrer Systemeinheit.

Zugehörige Konzepte:

„Kommunikationsadapter entfernen“ auf Seite 587

Bevor Sie einen Hot-Plug-Adapter entfernen oder austauschen können, müssen Sie diesen Adapter dekonfigurieren.

Zugehörige Tasks:

„Speicheradapter dekonfigurieren“ auf Seite 593

Bevor Sie einen Speicheradapter entfernen oder austauschen können, müssen Sie diesen Adapter dekonfigurieren.

„Asynchrone Adapter dekonfigurieren“ auf Seite 594

Sie können einen asynchronen Adapter dekonfigurieren.

Informationen zu PCI-Hot-Plug-Steckplätzen anzeigen

Bevor Sie einen Hot-Plug-Adapter hinzufügen, entfernen oder austauschen, können Sie Informationen zu den PCI-Hot-Plug-Steckplätzen in einer Maschine anzeigen.

Sie können die folgenden Informationen anzeigen:

- Liste aller PCI-Hot-Plug-Steckplätze in der Maschine
- Belegungsstatus des Steckplatzes: verfügbar oder leer
- Derzeit verwendete Steckplätze
- Merkmale eines bestimmten Steckplatzes, z. B. Steckplatznamen, Beschreibungen, Anschlusstyp und Name der angeschlossenen Einheit

Sie können SMIT oder Systembefehle verwenden. Zur Ausführung dieser Tasks müssen Sie als Root angemeldet sein.

Prozedur mit dem SMIT-Direktaufruf

1. Geben Sie an einer Eingabeaufforderung **smit devdrpci** ein, und drücken Sie anschließend die Eingabetaste.
2. Verwenden Sie die SMIT-Dialoge, um die Tasks auszuführen.

Wenn Sie zusätzliche Informationen für die Ausführung der Task benötigen, können Sie die Hilfetaste F1 in den SMIT-Dialogen auswählen.

Prozedur mit Befehlen

Mit den folgenden Befehlen können Sie Informationen über Hot-Plug-Steckplätze und angeschlossene Einheiten anzeigen:

- Der Befehl `lsslot` zeigt eine Liste aller PCI-Hot-Plug-Steckplätze und ihrer Merkmale an.
- Der Befehl `lsdev` zeigt den aktuellen Status aller Einheiten an, die im System installiert sind.

PCI-Kommunikationsadapter dekonfigurieren

Im Folgenden finden Sie eine Übersicht über die Dekonfiguration von PCI-Kommunikationsadaptern. Dazu gehören Ethernet-, Token-Ring-, FDDI- und ATM-Adapter.

Wenn Ihre Anwendung das Protokoll TCP/IP verwendet, müssen Sie die TCP/IP-Schnittstelle für den Adapter aus der Netzschnittstellenliste entfernen, bevor Sie den Adapter in den Status "Definiert" (Defined) versetzen können. Mit dem Befehl `netstat` können Sie feststellen, ob Ihr Adapter für TCP/IP konfiguriert ist, und die aktiven Netzschnittstellen an Ihrem Adapter überprüfen. Weitere Informationen finden Sie in der Beschreibung des Befehls `netstat` in der Veröffentlichung "Commands Reference".

Ein Ethernet-Adapter kann zwei Schnittstellen haben: Standard-Ethernet (enX) oder IEEE 802.3 (etX). Das X in enX und das X im Adapternamen (entX) haben denselben Wert. TCP/IP kann jeweils nur von einer dieser Schnittstellen verwendet werden. Der Ethernet-Adapter ent0 kann beispielsweise die Schnittstellen en0 und et0 haben.

Ein Token-Ring-Adapter kann nur eine Schnittstelle haben: Token-Ring (trX). Das X in trX und das X im Adapternamen (tokX) haben denselben Wert. Der Token-Ring-Adapter tok0 hat beispielsweise eine Schnittstelle tr0.

Ein ATM-Adapter kann nur eine ATM-Schnittstelle haben: ATM (atX). Das X in atX und das X im Adapternamen (atmX) haben denselben Wert. Der ATM-Adapter atm0 hat beispielsweise eine Schnittstelle at0. ATM-Adapter können jedoch mehrere emulierte Clients haben, die über einen einzigen Adapter ausgeführt werden.

Der Befehl `ifconfig` entfernt eine Schnittstelle aus dem Netz. Der Befehl `rmdev` dekonfiguriert die PCI-Einheit, behält aber die zugehörige Einheitendefinition in der Objektklasse für angepasste Einheiten. Sobald der Adapter den Status "Definiert" hat, können Sie mit dem Befehl `drslot` den Adapter entfernen.

Geben Sie Folgendes ein, um die untergeordneten Einheiten des PCI-Busses pci1 und deren untergeordnete Einheiten zu dekonfigurieren, aber die Einheitendefinitionen in der Objektklasse für angepasste Einheiten zu behalten:

```
rmdev -p pci1
```

Das System zeigt eine Nachricht wie die folgende an:

```
rmt0 Defined
hdisk1 Defined
scsil Defined
ent0 Defined
```

Zugehörige Konzepte:

„Kommunikationsadapter entfernen“ auf Seite 587

Bevor Sie einen Hot-Plug-Adapter entfernen oder austauschen können, müssen Sie diesen Adapter dekonfigurieren.

Einen PCI-Hot-Plug-Adapter entfernen

Sie können einen PCI-Hot-Plug-Adapter aus der Systemeinheit entfernen, ohne das Betriebssystem herunterzufahren oder das System ausschalten zu müssen. Wenn Sie einen Adapter entfernen, stehen die von diesem Adapter bereitgestellten Ressourcen dem Betriebssystem und den Anwendungen nicht mehr zur Verfügung.

Bevor Sie einen Adapter entfernen können, müssen Sie ihn dekonfigurieren.

Im Folgenden sind die Prozeduren für das Entfernen eines PCI-Hot-Plug-Adapters beschrieben. Sie können diese Tasks mit SMIT oder Systembefehlen ausführen. Zur Ausführung dieser Tasks müssen Sie als Root angemeldet sein.

Wenn Sie einen Adapter gegen einen anderen Adapter desselben Typs austauschen, werden die Konfigurationsdaten des ausgetauschten Adapters beibehalten und mit der Ersatzkarte verglichen. Der vorhandene Einheits-treiber des ausgetauschten Adapters muss den Ersatzadapter unterstützen.

Prozedur mit dem SMIT-Direktaufruf

1. Geben Sie an einer Eingabeaufforderung `smit devdrpci` ein, und drücken Sie anschließend die Eingabetaste.
2. Verwenden Sie die SMIT-Dialoge, um die Tasks auszuführen.

Wenn Sie zusätzliche Informationen für die Ausführung der Task benötigen, können Sie die Hilfetaste F1 in den SMIT-Dialogen auswählen.

Prozedur mit Befehlen

Sie können die folgenden Befehle verwenden, um Informationen über die Hot-Plug-Steckplätze und die angeschlossenen Einheiten anzuzeigen und einen PCI-Hot-Plug-Adapter zu entfernen:

- Der Befehl `lsslot` zeigt eine Liste aller PCI-Hot-Plug-Steckplätze und ihrer Merkmale an. Weitere Verwendungsinformationen finden Sie in der Beschreibung des Befehls `lsslot` in der Veröffentlichung *Commands Reference, Volume 3*.
- Der Befehl `lsdev` zeigt den aktuellen Status aller Einheiten an, die im System installiert sind. Weitere Verwendungsinformationen finden Sie in der Beschreibung des Befehls `lsdev` in der Veröffentlichung *Commands Reference, Volume 3*.
- Der Befehl `drslot` bereitet einen Hot-Plug-Steckplatz für das Entfernen eines Hot-Plug-Adapters vor. Weitere Verwendungsinformationen finden Sie in der Beschreibung des Befehls `drslot` in der Veröffentlichung *Commands Reference, Volume 2*.

Zugehörige Konzepte:

„Kommunikationsadapter entfernen“ auf Seite 587

Bevor Sie einen Hot-Plug-Adapter entfernen oder austauschen können, müssen Sie diesen Adapter dekonfigurieren.

Zugehörige Tasks:

„Speicheradapter dekonfigurieren“ auf Seite 593

Bevor Sie einen Speicheradapter entfernen oder austauschen können, müssen Sie diesen Adapter dekonfigurieren.

„Asynchrone Adapter dekonfigurieren“ auf Seite 594

Sie können einen asynchronen Adapter dekonfigurieren.

Einen PCI-Hot-Plug-Adapter hinzufügen

Sie können einen PCI-Hot-Plug-Adapter in einem verfügbaren Steckplatz in der Systemeinheit installieren und dem Betriebssystem und den Anwendungen neue Ressourcen zur Verfügung stellen, ohne das Betriebssystem erneut booten zu müssen. Sie können einen zusätzlichen Adapter desselben Typs oder einen Adapter eines anderen Typs installieren.

Im Folgenden sind die Prozeduren für das Hinzufügen eines neuen PCI-Hot-Plug-Adapters beschrieben.

Achtung: Bevor Sie versuchen, PCI-Hot-Plug-Adapter zu hinzuzufügen, lesen Sie das Handbuch *PCI Adapter Placement Reference*, das zu den Systemeinheiten geliefert wird, die die Hot-Plug-Technologie unterstützen, um festzustellen, ob der Adapter im laufenden Betrieb ausgetauscht werden kann. Anweisungen zum Installieren und Entfernen von Adaptern finden Sie in der Dokumentation zu Ihrer Systemeinheit.

Das Hinzufügen eines neuen PCI-Hot-Plug-Adapters umfasst die folgenden Tasks:

- Einen verfügbaren Steckplatz in der Maschine suchen und identifizieren
- Steckplatz für die Konfiguration des Adapters vorbereiten
- Einheitentreiber installieren, sofern erforderlich
- Neuen Adapter konfigurieren

Sie können SMIT oder Systembefehle verwenden. Zur Ausführung dieser Tasks müssen Sie als Root angemeldet sein.

Anmerkung: Wenn Sie dem System einen Hot-Plug-Adapter hinzufügen, können dieser Adapter und seine untergeordneten Einheiten möglicherweise nicht mit dem Befehl **bootlist** als Booteinheit angegeben werden. Unter Umständen müssen Sie einen Warmstart des Systems durchführen, um dem Betriebssystem alle potenziellen Booteinheiten bekannt zu machen.

Prozedur mit dem SMIT-Direktaufruf

1. Geben Sie an einer Eingabeaufforderung `smit devdrpci` ein, und drücken Sie anschließend die Eingabetaste.
2. Verwenden Sie die SMIT-Dialoge, um die Task auszuführen.

Wenn Sie zusätzliche Informationen für die Ausführung der Task benötigen, können Sie die Hilfetaste F1 in den SMIT-Dialogen auswählen.

Prozedur mit Befehlen

Sie können die folgenden Befehle verwenden, um Informationen über die PCI-Hot-Plug-Steckplätze und die angeschlossenen Einheiten anzuzeigen und einen PCI-Hot-Plug-Adapter hinzuzufügen:

- Der Befehl **lsslot** zeigt eine Liste aller Hot-Plug-Steckplätze und ihrer Merkmale an. Weitere Verwendungsinformationen finden Sie in der Beschreibung des Befehls **lsslot** in der Veröffentlichung *Commands Reference, Volume 3*.
- Der Befehl **lsdev** zeigt den aktuellen Status aller Einheiten an, die im System installiert sind. Weitere Verwendungsinformationen finden Sie in der Beschreibung des Befehls **lsdev** in der Veröffentlichung *Commands Reference, Volume 3*.
- Der Befehl **drslot** bereitet den Hot-Plug-Steckplatz für das Hinzufügen oder Entfernen eines Hot-Plug-Adapters vor. Weitere Verwendungsinformationen finden Sie in der Beschreibung des Befehls **drslot** in der Veröffentlichung *Commands Reference, Volume 2*.

Multiple Path I/O

Mit Multiple Path I/O (MPIO) kann eine Einheit über eine oder mehrere physische Verbindungen oder *Pfade* eindeutig erkannt werden.

Ein Pfadsteuerungsmodul (PCM, Path-Control Module) stellt die Pfadverwaltungsfunktionen bereit.

Ein MPIO-fähiger Einheitentreiber kann mehrere Typen von Zieleinheiten steuern. Ein PCM kann eine oder mehrere bestimmte Einheiten unterstützen. Deshalb kann ein Einheitentreiber mit mehreren PCMs kommunizieren, die die Ein-/Ausgabe über die Pfade zu jeder der Zieleinheiten steuern.

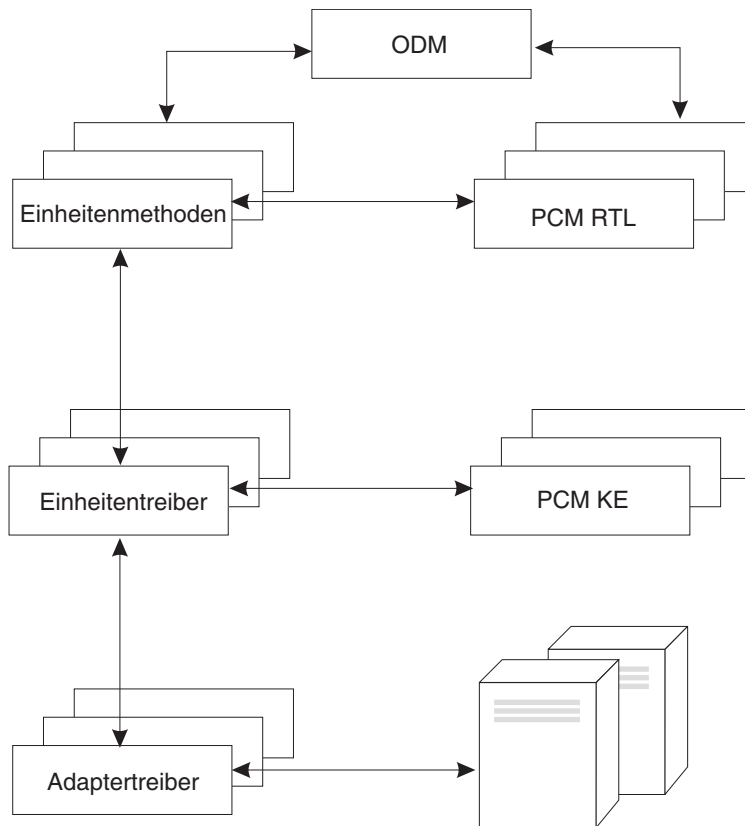


Abbildung 12. Interaktion von MPIO-Komponenten. Die folgende Abbildung zeigt die Interaktion zwischen den verschiedenen Komponenten, aus denen sich die MPIO-Lösung zusammensetzt. In dieser Abbildung steuert der MPIO-Einheitentreiber mehrere Typen von Zieleinheiten, die jeweils ein anderes PCM erfordern. (KE=Kernelerweiterung, RTL=Run-time Loadable, zur Laufzeit ladbare Komponente).

Bevor eine Einheit MPIO nutzen kann, müssen Treiber, Methoden und vordefinierte Attribute der Einheit im Object Data Manager (ODM) geändert werden, damit die Erkennung, Konfiguration und Verwaltung mehrerer Pfade unterstützt werden. Die Einheitentreiber für parallele SCSI- und Fibre-Channel-Platten und ihre Einheitenmethoden unterstützen MPIO-Platteneinheiten. Die iSCSI-Platteneinheiten werden als MPIO-Einheiten unterstützt. Der Einheitentreiber für Fibre-Channel-Bandlaufwerke und dessen Einheitenmethoden unterstützen MPIO-Bandeinheiten. Außerdem wurden die vordefinierten Attribute für einige Einheiten im ODM für MPIO geändert.

Das AIX-PCM setzt sich aus dem Konfigurationsmodul PCM RTL und der Kernelerweiterung PCM KE zusammen. Das PCM RTL ist ein Modul, das zur Laufzeit geladen werden kann und den Einheitenmethoden ermöglicht, zusätzliche einheitenspezifische oder ODM-Pfadattribute zu erkennen, die die PCM KE erfordert. Das PCM RTL wird von einer Einheitenmethode geladen. Anschließend werden Routinen im PCM RTL aufgerufen, um bestimmte Operationen auszuführen, die PM-KE-Variablen initialisieren oder ändern.

Die PCM KE stellt allen Einheitentreibern, die die MPIO-Schnittstelle unterstützen, Verwaltungsfunktionen für die Pfadsteuerung zur Verfügung. Die PCM KE ist davon abhängig, dass die Einheitenkonfiguration Pfade erkennt und diese Informationen an den Einheitentreiber überträgt. Jeder MPIO-fähige Einheitentreiber fügt die Pfade zu einer Einheit von den ihr direkt übergeordneten Einheiten hinzu. Die Wartung und Planung der Ein- und Ausgabe in den verschiedenen Pfaden wird von der PCM KE übernommen und ist für den MPIO-fähigen Einheitentreiber nicht ersichtlich.

Die PCM KE kann mehrere Weiterleitungsalgorithmen bereitstellen, zwischen denen der Benutzer wählen kann. Außerdem unterstützt die PCM KE die Erfassung von Informationen, auf deren Basis der beste

Pfad für eine E/A-Anforderung bestimmt und ausgewählt werden kann. Die PCM KE kann den besten Pfad nach verschiedenen Kriterien auswählen, z. B. Lastausgleich, Verbindungsgeschwindigkeit, Verbindungsfehler usw.

Das AIX-PCM hat eine Statusprüfungsfunktion, die für folgende Zwecke verwendet werden kann:

- Die Pfade prüfen und bestimmen, welche Pfade derzeit für das Senden von Ein-/Ausgaben verwendet werden können.
- Einen Pfad aktivieren, der zuvor aufgrund eines temporären Pfadfehlers (z. B. ein Kabel einer Einheit wurde entfernt und anschließend wieder angeschlossen) als fehlerhaft markiert wurde.
- Die derzeit nicht verwendeten Pfade überprüfen, die im Falle eines Failover verwendet werden würden (wenn der Wert für das Algorithmusattribut beispielsweise `failover` ist, kann die Statusprüfung die Alternativpfade testen).

Nicht alle Platteneinheiten und Bandeneinheiten können mit den AIX-Standard-PCMs erkannt und konfiguriert werden. Die AIX-Standard-PCMs umfassen zwei Pfadsteuerungsmodulare: ein Modul für die Verwaltung von Platteneinheiten und ein weiteres Modul für die Verwaltung von Bandeneinheiten. Wenn Ihre Einheit nicht erkannt wird, fragen Sie bei Ihrem Einheitenlieferanten nach, ob ein PCM für Ihre Einheit verfügbar ist.

Zugehörige Verweise:

„Attribute für andere SCSI-Bänder (Typ `ost`)“ auf Seite 611

Im Folgenden sind Attribute für andere SCSI-Bänder (Typ `ost`) beschrieben.

Verwaltung MPIO-fähiger Einheiten

Die Funktion Multiple Path I/O (MPIO) kann verwendet werden, um für Failover-Zwecke alternative Pfade zu einer Einheit zu definieren.

Failover ist ein Algorithmus für die Pfadverwaltung, der die Zuverlässigkeit und Verfügbarkeit einer Einheit verbessert, weil das System automatisch erkennt, wenn ein E/A-Pfad ausfällt, und in diesem Fall E/A-Anforderungen über einen alternativen Pfad umleitet. Alle SCSI-SCSD-Plattenlaufwerke werden automatisch als MPIO-Einheiten konfiguriert, und es kann eine ausgewählte Anzahl von Fibre-Channel-Plattenlaufwerken als "Andere MPIO-Platteneinheit" konfiguriert werden. Es können weitere Einheiten unterstützt werden, sofern der Einheitentreiber mit der MPIO-Implementierung in AIX kompatibel ist.

MPIO wird im Rahmen der Installation des Basisbetriebssystems installiert und konfiguriert. Es ist zwar keine weitere Konfiguration erforderlich, aber Sie können Einheiten mit SMIT oder der Befehlszeilenschnittstelle hinzufügen, entfernen, rekonfigurieren, aktivieren und inaktivieren. Die folgenden Befehle unterstützen Sie bei der Verwaltung von MPIO-Pfaden:

mkpath

Fügt einen Pfad zu einer Zieleinheit hinzu.

rmpath

Entfernt einen Pfad zu einer Zieleinheit.

chpath

Ändert ein Attribut oder den Betriebsstatus eines Pfades zu einer Zieleinheit.

lsmpio

Zeigt Informationen wie den Status, die Konfiguration und Statistiken zu MPIO-Speicherheiten (MultiPath I/O) an.

lspath Zeigt Informationen zu Pfaden zu einer Zieleinheit an.

Zugehörige Konzepte:

„Einheitenkonfigurationsdatenbank und Einheitenverwaltung“ auf Seite 564

Einheitendaten sind in einer vordefinierten oder angepassten Datenbank enthalten, die als Einheitenkonfigurationsdatenbank verwendet wird.

„MPIO-Einheit konfigurieren“ auf Seite 579

Für die Konfiguration einer MPIO-fähigen Einheit werden dieselben Befehle verwendet wie für Nicht-MPIO-Einheiten.

Eine SCSI-Einheit als MPIO-Einheit verkabeln:

Eine SCSI-Einheit kann von maximal zwei Adaptern unterstützt werden, wenn sie als MPIO-fähige Einheit konfiguriert ist.

Wenn Sie eine parallele SCSI-Einheit als MPIO-Einheit verkabeln möchten, verwenden Sie die folgende einfache Konfiguration als Beispiel. Im Folgenden wird die erforderliche Mindestkonfiguration beschrieben. Für Ihre Einheit sind möglicherweise weitere Konfigurationsschritte erforderlich.

1. Installieren Sie bei ausgeschaltetem System zwei SCSI-Adapter.
2. Schließen Sie die Einheit an beide SCSI-Adapter an.
3. Schalten Sie das System ein.
4. Ändern Sie die Einstellungen in einem der Adapter in eine eindeutige SCSI-ID. Standardmäßig haben SCSI-Adapter die SCSI-ID 7. Da jede ID eindeutig sein muss, müssen Sie die SCSI-ID eines Adapters ändern, z. B. in 6.
5. Führen Sie den Befehl `cfgmgr` aus.
6. Geben Sie zum Überprüfen der Konfiguration in der Befehlszeile den folgenden Befehl ein:

```
lspath -l hdiskX
```

Das X steht für die logische Nummer der neu konfigurierten Einheit. Die Befehlsausgabe sollte zwei Pfade und ihren Status enthalten.

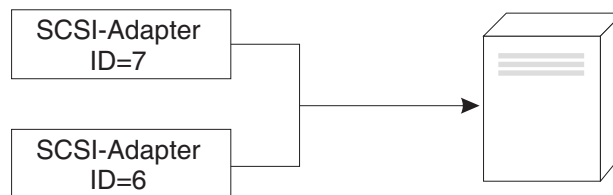


Abbildung 13. Kabelkonfiguration für MPIO-SCSI-Einheit

Diese Abbildung zeigt den Anschluss von zwei SCSI-Adaptern an dieselbe Einheit.

Eine Fibre-Channel-Einheit als MPIO-Einheit verkabeln:

Eine Fibre-Channel-Einheit kann an mehrere Adapter angeschlossen werden. In der Software ist keine Begrenzung definiert.

Wenn Sie eine Fibre-Channel-Einheit als MPIO-Einheit verkabeln möchten, verwenden Sie die folgende einfache Konfiguration als Beispiel. Im Folgenden wird die erforderliche Mindestkonfiguration beschrieben. Für Ihre Einheit sind möglicherweise weitere Konfigurationsschritte erforderlich.

1. Installieren Sie bei ausgeschaltetem System zwei Fibre-Channel-Adapter.
2. Schließen Sie die Adapter an einen Switch oder Hub an.
3. Schließen Sie die Einheit an den Switch bzw. Hub an.
4. Schalten Sie das System ein.
5. Geben Sie zum Überprüfen der Konfiguration in der Befehlszeile den folgenden Befehl ein:

```
lspath -l hdiskX
```

Das X steht für die logische Nummer der neu konfigurierten Einheit. Die Befehlsausgabe sollte für jeden installierten Adapter einen Pfad und den Status enthalten.

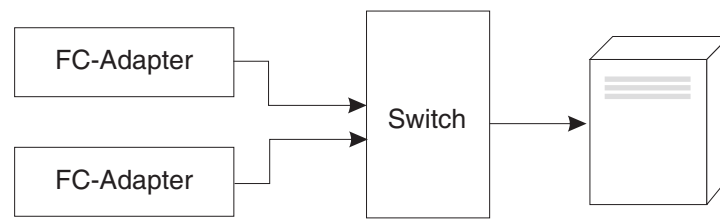


Abbildung 14. Kabelkonfiguration für MPIO-Fibre-Channel-Einheit. Diese Abbildung zeigt eine einfache Konfiguration mit zwei Fibre-Channel-Adaptoren an einem Switch, an den die Einheit angeschlossen ist.

MPIO-Einheit konfigurieren

Für die Konfiguration einer MPIO-fähigen Einheit werden dieselben Befehle verwendet wie für Nicht-MPIO-Einheiten.

Die Befehle **cfgmgr**, **mkdev**, **chdev**, **rmdev** und **lsdev** unterstützen die Verwaltung von MPIO-Einheiteninstanzen und zeigen deren Attribute an. Ein MPIO-Einheiteneintrag besitzt auch Pfadeinträge, die dem Einheiteneintrag zugeordnet sind. Die Befehle **mkpath**, **chpath**, **rmpath** und **lspath** verwalten Pfadeinträge und zeigen ihre Attribute an.

Ein Pfadeintrag für eine MPIO-Einheit kann hinzugefügt oder entfernt werden, ohne die Einheit dekonfigurieren zu müssen.

Eine MPIO-Einheit kann weitere Attribute haben, aber die erforderlichen Attribute, die alle MPIO-Einheiten unterstützen müssen, sind **reserve_policy** und **algorithm**. Das Attribut **reserve_policy** bestimmt den Typ der Reservierungsmethodik, die der Einheitentreiber beim Öffnen der Einheit implementiert, und kann verwendet werden, um den Zugriff auf die Einheit durch andere Adapter in demselben oder in einem anderen System zu beschränken. Das Attribut **algorithm** definiert die Methodik, die das PCM für die E/A-Verwaltung in den für eine Einheit konfigurierten Pfaden verwendet.

Zugehörige Konzepte:

„Verwaltung MPIO-fähiger Einheiten“ auf Seite 577

Die Funktion Multiple Path I/O (MPIO) kann verwendet werden, um für Failover-Zwecke alternative Pfade zu einer Einheit zu definieren.

„Attribute für MPIO-Einheiten“ auf Seite 581

Die folgenden Attribute werden nur von MPIO-Einheiten unterstützt. Die Attribute können mit SMIT oder Befehlen (insbesondere den Befehlen **lsattr** und **chdev**) angezeigt oder geändert werden.

„Attribute für Pfadsteuerungsmodule“ auf Seite 582

Zusätzlich zu den AIX-Standardpfadsteuerungsmodulen (PCM, Path Control Module) kann ein einheitenspezifisches PCM von einem Einheitenlieferanten zur Verfügung gestellt werden. Die Gruppe der vom Benutzer änderbaren Attribute wird vom Einheitenlieferanten definiert. Ein einheitenspezifisches PCM kann Einheiten- und Pfadattribute haben.

Unterstützte MPIO-Einheiten

Die AIX-Standard-PCMs unterstützen eine Gruppe von Platteneinheiten und Bänderheiten, die in der Dateigruppe `devices.common.IBM.mpio.rte` definiert sind.

Für Einheiten, die von den AIX-PCMs für Platteneinheiten oder den PCMs für Bänderheiten nicht unterstützt werden, muss der Einheitenlieferant vordefinierte Attribute im ODM, ein PCM und weiteren unterstützenden Code bereitstellen, der erforderlich ist, um die Einheit als MPIO-fähige Einheit zu erkennen.

Führen Sie das folgende Script aus, um festzustellen, welche Platteneinheiten vom AIX-PCM für Platteneinheiten unterstützt werden:

```
odmget -qDvDr=aixdiskpcmke PdDv | grep uniquetype | while read line
do
    utype=`echo $line | cut -d'"' -f2`
    dvc=`odmget -q"uniquetype=$utype AND attribute=dvc_support" PdAt`
    echo $dvc | grep values | cut -d'"' -f2
done
```

Führen Sie das folgende Script aus, um festzustellen, welche Platteneinheiten vom AIX-PCM für Bandeinheiten unterstützt werden:

```
odmget -qDvDr=aixtapepcmke PdDv | grep uniquetype | while read line
do
    utype=`echo $line | cut -d'"' -f2`
    dvc=`odmget -q"uniquetype=$utype AND attribute=dvc_support" PdAt`
    echo $dvc | grep values | cut -d'"' -f2
done
```

Die Scriptausgabe enthält eine Liste der eindeutigen Einheitentypen, die von den AIX-Standard-PCMs unterstützt werden. Die drei vom AIX-PCM für Platteneinheiten unterstützten Einheitentypen sind *selbst-konfigurierende parallele SCSI-Platte* (disk/scsi/scsd), *Andere FC-Platteneinheit mit MPIO* (disk/fcp/mpioosdisk) und *Andere iSCSI-Platteneinheit mit MPIO* (disk/iscsi/mpioosdisk). Der vom AIX-PCM für Bandeinheiten unterstützte Einheitentyp ist *Andere FC-Platteneinheit mit MPIO* (tape/fcp/mpioost).

Die Einheitentypen *Andere FC-Platteneinheit mit MPIO* und *Andere FC-Bandeinheit mit MPIO* sind Untergruppen von "Andere Fibre-Channel-Platten" bzw. "Andere Fibre-Channel-Bandeinheiten". Eine Einheit wird nur dann als *Andere FC-Einheit mit MPIO* unterstützt, wenn keine vom Einheitenlieferanten vordefinierten Attribute im ODM vorhanden sind und die Einheit für den Einsatz mit einem der AIX-Standard-PCMs zertifiziert wurde. Die Zertifizierung garantiert nicht, dass alle Einheitenfunktionen unterstützt werden, wenn die Einheit als *Andere FC-Einheit mit MPIO* konfiguriert ist.

Eine *Andere iSCSI-Platte mit MPIO* ist eine Untergruppe von "Andere iSCSI-Platten". Eine Einheit wird nur dann als *Andere iSCSI-Platte mit MPIO* unterstützt, wenn keine vom Einheitenlieferanten vordefinierten Attribute im ODM vorhanden sind und die Einheit für den Einsatz mit dem AIX-PCM zertifiziert wurde. Die Zertifizierung garantiert nicht, dass alle Einheitenfunktionen unterstützt werden, wenn die Einheit als *Andere iSCSI-Einheit mit MPIO* konfiguriert ist.

Führen Sie das folgende Script aus, um festzustellen, welche Einheiten als *Andere FC-Platten mit MPIO* unterstützt werden:

```
odmget -quniquetype=disk/fcp/mpioosdisk PdAt | grep deflt | cut -d'"' -f2
```

Führen Sie das folgende Script aus, um festzustellen, welche Einheiten als *Andere FC-Bandeinheiten mit MPIO* unterstützt werden:

```
odmget -q "uniquetype=tape/fcp/mpioosdisk AND attribute=mpio_model_map" PdAt | grep deflt | cut -d'"' -f2
```

Führen Sie das folgende Script aus, um festzustellen, welche Einheiten als *Andere iSCSI-Platten mit MPIO* unterstützt werden:

```
odmget -quniquetype=disk/iscsi/mpioosdisk PdAt | grep deflt | cut -d'"' -f2
```

Die Scriptausgabe enthält eine Liste der abgefragten Daten, zu denen der Einheitenlieferant und das Einheitenmodell gehören.

Führen Sie das folgende Script aus, um alle MPIO-fähigen Einheiten anzuzeigen, die im System installiert sind:

```
odmget -qattribute=unique_id PdAt | grep uniquetype | cut -d'"' -f2
```

Die Scriptausgabe enthält eine Liste der eindeutigen MPIO-fähigen Einheitentypen, die von den AIX-Standard-PCMs und den vom Lieferanten bereitgestellten PCMs unterstützt werden.

Attribute für MPIO-Einheiten

Die folgenden Attribute werden nur von MPIO-Einheiten unterstützt. Die Attribute können mit SMIT oder Befehlen (insbesondere den Befehlen **lsattr** und **chdev**) angezeigt oder geändert werden.

Einige MPIO-Einheitenattribute (Multiple Path I/O) unterstützen die gleichzeitig ablaufende Aktualisierung des Attributs. Sie können die Attributwerte aktualisieren, während die Platte geöffnet und im Gebrauch ist, und die neuen Werte werden sofort wirksam. Für einige Attribute, insbesondere für das Attribut **reserve_policy**, kann es Einschränkungen bezüglich des Zeitpunkts der Attributänderung oder bezüglich der vom Attribut akzeptierten neuen Werte geben. Wenn eine Platte beispielsweise geöffnet und momentan als Cluster-Repository-Platte genutzt wird, blockiert das Betriebssystem AIX alle Versuche, eine Reservierungsrichtlinie für die Platte festzulegen, weil dies dazu führen würde, dass andere Clusterknoten den Zugriff auf das Repository verlieren würden.

Das Einheitenattribut, das alle MPIO-Einheiten unterstützen müssen, ist **reserve_policy**. Gewöhnlich hat eine MPIO-Einheit auch das Einheitenattribut **PR_key**. Eine MPIO-Einheit kann weitere einheitenspezifische Attribute haben. Im Folgenden werden weitere einheitenspezifische Attribute beschrieben:

FC3_REC

Gibt an, ob die Einheit die Fehlerbehebung aktivieren muss, die Fibre-Channel-Klasse 3 verwendet. Die Aktivierung dieses Features verbessert die Fehlererkennung und Fehlerbehebung für bestimmte Typen von Fabricfehlern in Verbindung mit Fibre Channel. Dieses Attribut ist nur für eine begrenzte Gruppe von Einheiten verfügbar. Für dieses Attribut können die folgenden Werte definiert werden:

true Aktiviert die Fehlerbehebung, die die Fibre-Channel-Klasse 3 verwendet.

false Inaktiviert die Fehlerbehebung, die die Fibre-Channel-Klasse 3 verwendet.

reserve_policy

Definiert, ob beim Öffnen der Einheit eine Reservierungsmethodik verwendet wird. Die Werte sind wie folgt:

no_reserve

Es wird keine Reservierungsmethodik für die Einheit angewendet. Auch andere Initiatoren können, unter Umständen von anderen Hostsystemen, auf die Einheit zugreifen.

single_path_reserve

Es wird eine SCSI2-Reservierungsmethodik für die Einheit angewendet, d. h., nur der Initiator, der die Reservierungsanforderung abgesetzt hat, kann auf die Einheit zugreifen. Diese Strategie verhindert, dass andere Initiatoren auf demselben Host oder auf anderen Hosts auf die Einheit zugreifen. Bei dieser Strategie wird die SCSI2-Reservierungsmethodik verwendet, um die Einheit für einen einzelnen Initiator (Pfad) zu reservieren. Befehle, die über einen anderen Pfad weitergeleitet werden, führen zu einem Reservierungskonflikt.

Pfadauswahlalgorithmen, die Befehle über mehrere Pfade senden, können zu einer Überlastung führen, wenn der Wert **single_path_reserve** ausgewählt ist. Nehmen Sie an, ein einheitenspezifisches PCM hat ein erforderliches Attribut, dessen Wert dafür sorgt, dass die Ein-/Ausgabe auf mehrere Pfade verteilt wird. Wenn **single_path_reserve** wirksam ist, muss der Einheitentreiber ein Bussignal für Einheitenrücksetzung (BDR, Bus Device Reset) und anschließend eine Reservierungsanforderung absetzen und dabei für das Senden des nächsten Befehls einen neuen Pfad verwenden, um die vorherige Reservierung aufzuheben. Jedes Mal, wenn ein anderer Pfad ausgewählt wird, kommt es aufgrund des Aufwands für das Senden eines BDR und das Absetzen einer Reservierungsanforderung an die Zieleinheit zu einer Überlastung und zu Leistungseinbußen. (Die Auswahl eines Algorithmus, der zu einer Überlastung führen kann, wird vom AIX-PCM nicht zugelassen.)

PR_exclusive

Beim Öffnen der Einheit wird eine SCSI3-Methodik für persistente Reservierung und exklusiven Hostzugriff angewendet. Der Wert des Attributs **PR_key** muss für jedes Hostsystem eindeutig sein. Das Attribut **PR_key** wird verwendet, um den Zugriff auf die Einheit durch Initiatoren von anderen Hostsystemen zu verhindern.

PR_shared

Beim Öffnen der Einheit wird eine SCSI3-Methodik für persistente Reservierung und gemeinsamen Hostzugriff angewendet. Der Wert des Attributs **PR_key** muss für jedes Hostsystem eindeutig sein. Initiatoren von anderen Hostsystemen müssen sich registrieren, bevor sie auf die Einheit zugreifen dürfen.

PR_key

Dieses Attribut ist nur erforderlich, wenn die Einheit eine der Strategien für persistente Reservierung (**PR_exclusive** oder **PR_shared**) unterstützt.

Zugehörige Konzepte:

„MPIO-Einheit konfigurieren“ auf Seite 579

Für die Konfiguration einer MPIO-fähigen Einheit werden dieselben Befehle verwendet wie für Nicht-MPIO-Einheiten.

Zugehörige Informationen:

lsattr command

chdev command

Attribute für Pfadsteuerungsmodule

Zusätzlich zu den AIX-Standardpfadsteuerungsmodulen (PCM, Path Control Module) kann ein einheiten-spezifisches PCM von einem Einheitenlieferanten zur Verfügung gestellt werden. Die Gruppe der vom Benutzer änderbaren Attribute wird vom Einheitenlieferanten definiert. Ein einheitenspezifisches PCM kann Einheiten- und Pfadattribute haben.

Im Folgenden sind die Einheitenattribute für die AIX-Standard-PCMs beschrieben:

algorithm

Bestimmt die Methodik, mit der die Ein-/Ausgabe auf die Pfade für eine Einheit verteilt wird. Die gültigen Werte für dieses Attribut sind:

Anmerkung: Einige Einheiten unterstützen nur einen Teil dieser Werte.

failover

Sendet alle Ein-/Ausgaben über einen einzigen Pfad. Wenn der Pfad als fehlerhaft oder inaktiviert markiert ist, wird der nächste verfügbare Pfad ausgewählt, über den alle E/A-Operationen gesendet werden. Dieser Algorithmus verwaltet alle aktivierten Pfade in einer geordneten Liste basierend auf den aufsteigenden Werten des Attributs **Pfadpriorität**. Für jede E/A-Operation wird der gültige Pfad mit dem niedrigsten Pfadprioritätswert ausgewählt.

round_robin

Verteilt die E/A-Operationen auf mehrere aktivierte Pfade. Für Einheiten, die aktive und passive Pfade oder bevorzugte und nicht bevorzugte Pfade haben, wird nur ein Teil der Pfade für E/A-Operationen verwendet. Wenn ein Pfad als fehlerhaft oder inaktiviert markiert ist, wird er nicht mehr für das Senden von E/A-Operationen verwendet. Die E/A-Operation wird basierend auf dem Attribut **Pfadpriorität** verteilt. Pfade mit einem höheren Pfadprioritätswert erhalten einen höheren Anteil der E/A-Operationen.

shortest_queue

Verteilt die E/A-Operationen auf mehrere aktivierte Pfade. Für Einheiten, die aktive und passive Pfade oder bevorzugte und nicht bevorzugte Pfade haben, wird nur ein Teil der Pfade für E/A-Operationen verwendet. Dieser Algorithmus gleicht dem Algorithmus **round_robin**. Der Algorithmus **shortest_queue** verteilt die E/A-Operationen jedoch ba-

sierend auf der Anzahl anstehender E/A-Operationen in jedem Pfad. Der Pfad mit der derzeit niedrigsten Anzahl anstehender E/A-Operationen wird für die nächste Operation ausgewählt. Das Attribut **path priority** wird ignoriert, wenn der Algorithmus auf **shortest_queue** gesetzt ist.

hcheck_mode

Bestimmt, welche Pfade geprüft werden müssen, wenn die Funktion für Statusprüfung verwendet wird. Das Attribut unterstützt die folgenden Modi:

enabled permit rw- u:dhs

Sendet den Befehl **healthcheck** über die Pfade mit dem Status "enabled" (aktiviert). In diesem Modus wird der Befehl **healthcheck** nicht über Pfade mit dem Status "disabled" (inaktiviert) oder "missing" (fehlt) gesendet.

failed Sendet den Befehl **healthcheck** über die Pfade mit dem Status "failed" (fehlerhaft). In diesem Modus wird der Befehl **healthcheck** nicht über Pfade mit dem Status "enabled" (aktiviert), "disabled" (inaktiviert) oder "missing" (fehlt) gesendet.

nonactive

(Standardeinstellung) Sendet den Befehl **healthcheck** über Pfade, die keine aktive E/A-Verbindung zur Einheit haben, einschließlich der Pfade mit dem Status "failed" (fehlerhaft) oder "enabled" (aktiviert). In diesem Modus wird der Befehl **healthcheck** nicht über Pfade mit dem Status "disabled" (inaktiviert) oder "missing" (fehlt) gesendet.

hcheck_interval

Definiert, wie oft die Statusprüfung für die Pfade einer Einheit durchgeführt wird. Das Attribut unterstützt einen Bereich von 0 bis 3600 Sekunden. Bei Auswahl des Wertes 0 ist die Statusprüfung inaktiviert.

Anmerkung: Die Statusprüfung wird nur durchgeführt, wenn die Platte von einem Prozess geöffnet und noch nicht geschlossen wurde. Wenn keine Entität die Platte geöffnet hat, überprüft das Pfadsteuerungsmodul die Pfade selbst dann nicht, wenn das Attribut **hcheck_interval** dieser Einheit auf einen Wert ungleich null gesetzt ist.

dist_tw_width

Definiert die Dauer eines "Zeitfensters". In diesem Zeitfenster sammelt der Algorithmus für verteilte Fehlererkennung die E/As, die mit einem Fehler zurückgegeben werden. Die Maßeinheit für den Wert des Attributs **dist_tw_width** sind Millisekunden. Je niedriger der Attributwert ist, desto kleiner ist das Zeitfenster für die einzelnen Stichproben und desto weniger sensibel reagiert der Algorithmus auf kleine Häufungen von E/A-Fehlern. Ein höherer Attributwert erhöht die Sensitivität des Algorithmus auf kleine Anhäufungen von Fehlern und die Wahrscheinlichkeit, dass ein Pfad als fehlerhaft erkannt wird.

dist_err_percent

Definiert den Prozentsatz der "Zeitfenster", in denen ein Fehler in einem Pfad zulässig ist, bevor der Pfad aufgrund mangelhafter Leistung als fehlerhaft erkannt wird. Das Attribut **dist_err_percent** hat einen gültigen Wertebereich von 0 bis 100. Der Algorithmus für verteilte Fehlererkennung wird inaktiviert, wenn das Attribut auf null (0) gesetzt wird. Die Standardeinstellung ist null. Der Algorithmus für verteilte Fehlererkennung überprüft die Struktur, die die Einheit mit dem Adapter verbindet, auf Fehler. Der Algorithmus berechnet den Prozentsatz der Stichproben mit Fehlern und kennzeichnet einen Pfad als fehlerhaft, wenn der berechnete Wert größer ist als der Wert des Attributs **dist_err_percent**.

Im Folgenden wird das Pfadattribut für das AIX-PCM beschrieben:

path priority

Ändert das Verhalten der Algorithmusmethodik für die Liste der Pfade.

Wenn das Attribut "algorithm" den Wert "failover" hat, werden die Pfade in einer Liste verwaltet. Die Reihenfolge der Einträge in dieser Liste bestimmt, welcher Pfad zuerst ausgewählt wird und

welcher Pfad als nächster ausgewählt wird, falls ein Pfad fehlerhaft ist. Die Reihenfolge der Einträge wird durch den Wert des Attributs "path priority" bestimmt. Der Wert 1 steht für die höchste Priorität. Mehrere Pfade können zwar denselben Prioritätswert haben, aber wenn alle Pfade denselben Wert haben, basiert die Auswahl darauf, wann jeder einzelne Pfad konfiguriert wurde.

Wenn das Algorithmusattribut den Wert **round_robin** hat, weist der Algorithmus **path priority** jedem Pfad einen Prioritätswert zu. Die Pfade für die E/A-Operationen werden proportional zu den Pfadprioritäten ausgewählt. Deshalb werden Pfade mit höheren Prioritätswerten für mehr E/A-Operationen ausgewählt. Wenn alle Pfadprioritäten gleich sind, werden die Operationen gleichmäßig auf die Pfade verteilt.

cntl_hcheck_int

Die Statusprüffolge des Controllers wird gestartet, nachdem ein Fehler beim Transport einer Speicherstruktur erkannt wurde. Das Attribut **cntl_delay_time** bestimmt die maximal zulässige Dauer (in Sekunden) der aktiven Statusprüffolge des Controllers. Wenn ein Statusprüfbefehl des Controllers erfolgreich ausgeführt und ein verfügbarer Pfad gefunden wird, wird die Statusprüffolge des Controllers beendet, woraufhin die Eingabe/Ausgabe fortgesetzt werden kann. Wenn am Ende der Statusprüffolge des Controllers keine gültigen Pfade gefunden wurden, schlagen alle anstehenden und nachfolgenden E/A-Aktivitäten für die Einheit fehl, bis das Statusprüfprogramm der Einheit feststellt, dass ein fehlgeschlagener Pfad zurückgegeben wurde.

Während die Statusprüffolge des Controllers aktiv ist, gibt das Attribut **cntl_hcheck_interval** den Zeitraum (in Sekunden) an, nach dem die nächste Gruppe von Statusprüfbefehlen des Controllers abgesetzt werden. Der Wert des Attributs **cntl_hcheck_interval** muss kleiner als der Wert des Attributs **cntl_delay_time** sein, sofern das Attribut nicht auf 0 oder gar nicht gesetzt wird. Wenn **cntl_delay_time** oder **cntl_hcheck_interval** auf 0 gesetzt ist, ist dieses Feature inaktiviert.

cntl_delay_time

Die Statusprüffolge des Controllers wird gestartet, nachdem ein Fehler beim Transport einer Speicherstruktur erkannt wurde. Das Attribut **cntl_delay_time** bestimmt die maximal zulässige Dauer (in Sekunden) der Statusprüffolge des Controllers. Wenn ein Statusprüfbefehl des Controllers erfolgreich ausgeführt und ein verfügbarer Pfad gefunden wird, wird die Statusprüffolge des Controllers beendet, woraufhin die Eingabe/Ausgabe fortgesetzt werden kann. Wenn am Ende der Statusprüffolge des Controllers keine gültigen Pfade gefunden wurden, schlagen alle anstehenden und nachfolgenden E/A-Aktivitäten für die Einheit fehl, bis das Statusprüfprogramm der Einheit feststellt, dass ein fehlgeschlagener Pfad zurückgegeben wurde.

Während die Statusprüffolge des Controllers aktiv ist, gibt das Attribut **cntl_hcheck_interval** den Zeitraum (in Sekunden) an, nach dem die nächste Gruppe von Statusprüfbefehlen des Controllers abgesetzt werden. Der Wert des Attributs **cntl_hcheck_interval** muss kleiner als der Wert des Attributs **cntl_delay_time** sein, sofern das Attribut nicht auf 0 oder gar nicht gesetzt wird. Wenn **cntl_delay_time** oder **cntl_hcheck_interval** auf 0 gesetzt ist, ist dieses Feature inaktiviert.

timeout_policy

Passt das PCM-Verhalten bei Überschreitungen des Befehlszeitlimits und Transportfehlern an. Wenn **timeout_policy** auf **fail_path** oder **disable_path** gesetzt wird, können Leistungseinbußen aufgefangen werden, wenn auf einer MPIO-Einheit (Multiple Path I/O) (MPIO) temporäre SAN-Fabric-Probleme bei einigen Pfaden zur Einheit auftreten. Die folgenden Werte können für das Attribut **timeout_policy** angegeben werden:

retry_path

Das erste Auftreten einer Überschreitung des Befehlszeitlimits hat keinen sofortigen Pfausfall zur Folge. Wenn ein aufgrund von Transportproblemen ausgefallener Pfad durch eine Statusprüfung wiederhergestellt wird, kann der wiederhergestellte Pfad sofort wiederverwendet werden.

fail_path

Der Pfad fällt beim ersten Auftreten einer Überschreitung des Befehlszeitlimits aus, sofern es sich nicht um den letzten Pfad in der Pfadgruppe handelt. Wenn ein aufgrund von

Transportproblemen ausgefallener Pfad wiederhergestellt wird, wird der Pfad erst nach Ablauf eines gewissen Zeitintervalls ohne Ausfälle in diesem Pfad für E/A-Lese- und -Schreiboperationen wiederverwendet. Wenn dieses Feature aktiviert ist, kann es vor der Weiterleitung von E/A-Lese- und -Schreiboperationen an Pfade, die nach einem Transportfehler wiederhergestellt wurden, zu einer Verzögerung kommen.

disable_path

Der Pfad fällt beim ersten Auftreten einer Überschreitung des Befehlszeitlimits aus, sofern es sich nicht um den letzten Pfad in der Pfadgruppe handelt. Wenn ein aufgrund von Transportproblemen ausgefallener Pfad wiederhergestellt wird, wird der Pfad erst nach Ablauf eines gewissen Zeitintervalls ohne Ausfälle in diesem Pfad für E/A-Lese- und -Schreiboperationen wiederverwendet. Treten in diesem Pfad in einem bestimmten Zeitintervall weiterhin Überschreitungen des Befehlszeitlimits auf, kann der Pfad inaktiviert werden. Inaktivierte Pfad bleiben inaktiviert (und damit nicht verwendbar), bis Sie eine der folgenden Aktionen ausführen: Ausführung des Befehls **chpath** zum Aktivieren des inaktivierten Pfad, Rekonfiguration der betroffenen Platte oder Warmstart des Systems.

Zugehörige Konzepte:

„MPIO-Einheit konfigurieren“ auf Seite 579

Für die Konfiguration einer MPIO-fähigen Einheit werden dieselben Befehle verwendet wie für Nicht-MPIO-Einheiten.

Attribute für die SAN-Replikation

AIX Multiple Path I/O (MPIO) muss installiert sein, und die Einheit muss das AIX-Pfadsteuerungsmodul (PCM, Path Control Module) verwenden. Diese Attribute sind von Einstellungen und Features abhängig, die vom Speichersubsystem bereitgestellt werden.

Die folgenden AIX-Attribute beziehen sich auf das Verhalten der LUNs (Logical Unit Number, Nummer der logischen Einheit), die über die Speichersubsysteme repliziert werden. Diese Attribute werden möglicherweise nicht von allen Speichersubsystemen und Mikrocodeversionen unterstützt. Clustering-Software und Software für hohe Verfügbarkeit, wie z. B. PowerHA SystemMirror, wird installiert, um das Management der SAN-Replikation (Storage Area Network, Speicherbereichsnetz) auf den Knoten in einem Cluster zu koordinieren. Die folgenden Attribute können in einem Virtual I/O Server (VIOS) nicht geändert werden.

san_rep_cfg

Bestimmt, wie eine synchrone Einheit, die Peer-to-Peer Remote Copy (PPRC) verwendet, im Betriebssystem AIX definiert und konfiguriert wird. Die eindeutige_ID der Platteninstanz wird von diesem Attribut beeinflusst und kann sich ändern, wenn der Attributwert geändert wird. Das Attribut **san_rep_cfg** ändert den Status der PPRC-Einheit im Speichersubsystem nicht. Die folgenden Werte können für das Attribut angegeben werden:

none [Default]

Konfiguriert die LUNs in einer synchronen Einheit, die PPRC als gesonderte logische Platteninstanzen verwendet.

new Definiert und konfiguriert die synchrone Einheit, die PPRC als einzelne logische Instanz verwendet. Die Einheit wird nur definiert und konfiguriert, wenn keine vorhandenen Platteninstanzen mit LUNs in der PPRC-Einheit übereinstimmen.

new_and_existing

Definiert und konfiguriert die synchrone Einheit, die PPRC als einzelne logische Instanz verwendet. Wenn keine logische Platteninstanz die PPRC-Einheit darstellt, wird eine neue Platteninstanz definiert.

migrate_disk

Definiert und konfiguriert die synchrone Einheit, die PPRC als einzelne Hdisk-Instanz verwendet, und verwendet die ausgewählte logische Platteninstanz für die Einheit. Die Operation wird nur auf Einheiten unterstützt, für die das Attribut **san_rep_device** auf "supported" oder "detected" gesetzt ist. Wenn das Attribut **san_rep_device** der Zieleinheit

auf "supported" gesetzt ist, funktioniert die Operation nur, wenn die SAN-Replikation nach der letzten Konfiguration der Platte im Speicher konfiguriert wurde. Diese Operation wird auf Platten unterstützt, die geöffnet sind und vom Betriebssystem AIX verwendet werden, wenn die Einheit nicht als Repository-Platte in einem Cluster verwendet wird. Die eindeutige_ID für die betroffene Platte wird aktualisiert, um die ID der PPRC-Einheit widerzuspiegeln.

revert_disk

Konfiguriert eine vorhandene synchrone Einheit, die eine logische Platteninstanz mit PPRC-Unterstützung für eine HDisk-Instanz der Einheit ohne PPRC-Unterstützung verwendet. Diese Operation wird nur auf Einheiten verwendet, deren Attribut **san_rep_device** auf "yes" gesetzt ist. Der Name der logischen Einheit und die Sonderdatei der Zielplatteninstanz bleiben unverändert. Die primäre (Quellen-)LUN für eine SAN-Replikationseinheit wird für die zurückgesetzte Hdisk-Instanz verwendet. Wenn die primäre (Quellen-)LUN nicht gefunden wird oder dem Host nicht bekannt ist, wird die sekundäre (Ziel-)LUN für die zurückgesetzte HDisk-Instanz verwendet. Diese Operation wird auf Platten unterstützt, die geöffnet sind und vom Betriebssystem AIX verwendet werden, wenn die Einheit nicht als Repository-Platte in einem Cluster verwendet wird. Die eindeutige_ID für die betroffene Platte wird aktualisiert, um die LUN-ID widerzuspiegeln.

Die Werte **none**, **new** und **new_and_existing** bedeuten, dass das Verhalten für alle Einheiten desselben eindeutigen ODM-Typs (Object Data Manager) aktualisiert wird. Der Befehl **chdef** wird verwendet, um die Werte **none**, **new** und **new_and_existing** zu setzen. Der Befehl **chdef** aktualisiert den Standardwert eines Attributs für alle Einheiten des angegebenen eindeutigen ODM-Typs. Der Befehl **chdef** aktualisiert auch die Attributwerte für die vorhandenen Einheiteninstanzen, die bereits definiert sind. Für Einheiten, die bei der Ausführung des Befehls **chdef** bereits konfiguriert sind, wird die Änderung erst wirksam, wenn ein Warmstart des System durchgeführt wird oder wenn diese Einheiten dekonfiguriert und anschließend neu konfiguriert werden. Der Befehl **chdef** benötigt die Klasse, die Unterklasse und den Typ des Attributs. Zum Bestimmen des eindeutigen ODM-Typs des Attributs **san_rep_cfg** verwenden Sie den Befehl `lsattr -l hdisk# -F class,subclass,type`. Beispiel:

```
lsdev -l hdisk0 -F class,subclass,type
disk,fcpl,aixmpiods8k
chdef -a san_rep_cfg=none -c disk -s fcp -t aixmpiods8k
chdef -a san_rep_cfg=new -c disk -s fcp -t aixmpiods8k
chdef -a san_rep_cfg=new_and_existing -c disk -s fcp -t aixmpiods8k
```

Die Werte **migrate_disk** und **revert_disk** bedeuten, dass das Verhalten für eine einzelne und bestimmte Einheiteninstanz aktualisiert wird. Der Befehl **chdev** muss verwendet werden, um den Wert **migrate_disk** oder **revert_disk** für eine bestimmte Einheit zu setzen. Der Befehl **chdev** aktualisiert den Wert nur für die angegebene Einheit. Beispiel:

```
chdev -a san_rep_cfg=migrate_disk -l hdisk0
chdev -a san_rep_cfg=revert_disk -l hdisk0
```

san_rep_device

Gibt an, dass die logische Platteninstanz als SAN-Replikationseinheit definiert ist. Dieses Attribut wird während der Plattenkonfiguration gesetzt und kann veraltet sein, wenn sich der Status der Einheit nach der Konfiguration geändert hat. Die folgenden Werte können für das Attribut angegeben werden:

no Die Einheit ist im Betriebssystem AIX nicht als SAN-Replikationseinheit konfiguriert. Diese Einheit erfüllt die Anforderungen für die Konfiguration als SAN-Replikationseinheit nicht.

supported

Die Einheit ist im Betriebssystem AIX nicht als SAN-Replikationseinheit konfiguriert. Die-

se Einheit erfüllt die Anforderungen für die Konfiguration als SAN-Replikationseinheit. Die Einheit ist jedoch momentan nicht als SAN-Replikationseinheit im Speichersubsystem konfiguriert.

detected

Die Einheit ist im Betriebssystem AIX nicht als SAN-Replikationseinheit konfiguriert. Das Betriebssystem AIX hat festgestellt, dass diese Einheit die Anforderungen für die Konfiguration als SAN-Replikationseinheit erfüllt und momentan als solche im Speichersubsystem konfiguriert ist.

yes Die Einheit ist im Betriebssystem AIX als SAN-Replikationseinheit konfiguriert.

Zugehörige Informationen:

chdef command

chdev command

lsdev command

Kommunikationsadapter entfernen

Bevor Sie einen Hot-Plug-Adapter entfernen oder austauschen können, müssen Sie diesen Adapter dekonfigurieren.

Die Dekonfiguration eines Kommunikationsadapters umfasst die folgenden Tasks:

- Alle Anwendungen schließen, die den Adapter verwenden, den Sie entfernen oder austauschen möchten
- Sicherstellen, dass alle an den Adapter angeschlossenen Einheiten identifiziert und gestoppt werden
- Alle Steckplätze, die derzeit im Gebrauch sind, oder einen Steckplatz auflisten, in dem ein bestimmter Adapter installiert ist
- Steckplatzposition des Adapters ermitteln
- Schnittstelleninformationen aus der Netzschnittstellenliste anzeigen und entfernen
- Adapter als nicht verfügbar kennzeichnen

Wenn Sie den Kommunikationsadapter mit den folgenden Prozeduren dekonfigurieren möchten, müssen Sie sich als **Root** anmelden:

Zugehörige Konzepte:

„Hot-Plug-Verwaltung von PCI-Einheiten“ auf Seite 570

Sie können einen neuen PCI-Hot-Plug-Adapter in einem verfügbaren PCI-Steckplatz installieren, während das Betriebssystem aktiv ist.

Zugehörige Tasks:

„PCI-Kommunikationsadapter dekonfigurieren“ auf Seite 573

Im Folgenden finden Sie eine Übersicht über die Dekonfiguration von PCI-Kommunikationsadaptern. Dazu gehören Ethernet-, Token-Ring-, FDDI- und ATM-Adapter.

„Einen PCI-Hot-Plug-Adapter entfernen“ auf Seite 573

Sie können einen PCI-Hot-Plug-Adapter aus der Systemeinheit entfernen, ohne das Betriebssystem herunterzufahren oder das System ausschalten zu müssen. Wenn Sie einen Adapter entfernen, stehen die von diesem Adapter bereitgestellten Ressourcen dem Betriebssystem und den Anwendungen nicht mehr zur Verfügung.

Ethernet-, Token-Ring-, FDDI- und ATM-Adapter dekonfigurieren:

Führen Sie die folgenden Schritte aus, um einen Ethernet-, Token-Ring-, FDDI- oder ATM-Adapter zu dekonfigurieren:

1. Geben Sie `lsslot -c pci` ein, um alle Hot-Plug-Steckplätze in der Systemeinheit aufzulisten und ihre Merkmale anzuzeigen.

2. Geben Sie den entsprechenden, in den folgenden Beispielen gezeigten SMIT-Befehl ein, um installierte Adapter aufzulisten und den aktuellen Status aller Einheiten in der Systemeinheit anzuzeigen:

| Eintrag | Beschreibung |
|--------------|------------------------------|
| smit lsdenet | Ethernet-Adapter auflisten |
| smit lsdtok | Token-Ring-Adapter auflisten |
| smit ls_atm | ATM-Adapter auflisten |

Die folgende Namenskonvention wird für die unterschiedlichen Adaptertypen verwendet:

| Eintrag
Name | Beschreibung
Adaptertyp |
|-----------------|----------------------------|
| atm0, atm1, ... | ATM-Adapter |
| ent0, ent1, ... | Ethernet-Adapter |
| tok0, tok1, ... | Token-Ring-Adapter |

3. Schließen Sie alle Anwendungen, die den Adapter verwenden, den Sie dekonfigurieren möchten. Zum Fortsetzen dieser Prozedur müssen die Netzspeicherauszugsadressen auf dem System inaktiviert werden. Gehen Sie wie folgt vor, um die Netzspeicherauszugsadressen zu suchen und zu inaktivieren:

- a. Geben Sie in einer Befehlszeile Folgendes ein:

```
smit dump
```

- b. Wählen Sie **Aktuelle Speicherauszugseinheiten anzeigen** aus.

- c. Prüfen Sie, ob für eine der konfigurierten Speicherauszugseinheiten eine Netzadresse ausgewiesen wird. Wenn nicht, beenden Sie SMIT und fahren mit Schritt 4 fort. Wenn Sie die Adresse einer Speicherauszugseinheit in eine lokale Adresse ändern möchten, wählen Sie **Abbrechen** aus, oder drücken Sie die Taste F3, um mit dem nächsten Schritt fortzufahren.

- d. Wenn für die primäre Speicherauszugseinheit eine Netzadresse angezeigt wird, ändern Sie sie in eine lokale Adresse. Wählen Sie dazu **Primäre Speicherauszugseinheit ändern** aus, und geben Sie anschließend im Feld **Primäre Speicherauszugseinheit** die lokale Adresse ein.

- e. Wenn für die sekundäre Speicherauszugseinheit eine Netzadresse angezeigt wird, ändern Sie sie in eine lokale Adresse. Wählen Sie dazu **Sekundäre Speicherauszugseinheit ändern** aus, und geben Sie anschließend im Feld **Sekundäre Speicherauszugseinheit** die lokale Adresse ein.

- f. Klicken Sie abschließend auf **OK**, oder drücken Sie die Eingabetaste.

4. Geben Sie `netstat -i` ein, um eine Liste aller konfigurierten Schnittstellen anzuzeigen und festzustellen, ob Ihr Adapter für TCP/IP konfiguriert ist. Es erscheint eine Ausgabe wie die folgende:

```
Name  Mtu   Network  Address          Ipkts  Ierrs  Opkts  Oerrs  Coll
lo0   16896 link#1    0.0.0.0          076    0      118    0      0
lo0   16896 127      127.0.0.1        076    0      118    0      0
lo0   16896 :::1     0.0.0.0          076    0      118    0      0
tr0   1492  link#2    8.0.5a.b8.b.ec   151    0      405    11     0
tr0   1492 19.13.97 19.13.97.106     151    0      405    11     0
at0   9180  link#3    0.4.ac.ad.e0.ad  0      0      0      0      0
at0   9180 6.6.6     6.6.6.5          0      0      0      0      0
en0   1500 link#5    0.11.0.66.11.1  212    0      1      0      0
en0   1500 8.8.8     8.8.8.106       212    0      1      0      0
```

Token-Ring-Adapter können nur eine Schnittstelle haben, Ethernet-Adapter zwei. ATM-Adapter können mehrere Schnittstellen haben.

5. Geben Sie den entsprechenden Befehl `ifconfig`, wie in den folgenden Beispielen gezeigt, ein, um die Schnittstelle aus der Netzschnittstellenliste zu entfernen.

| Eintrag | Beschreibung |
|---------------------|---|
| ifconfig en0 detach | Standard-Ethernet-Schnittstelle entfernen |
| ifconfig et0 detach | IEEE-802.3-Ethernet-Schnittstelle entfernen |
| ifconfig tr0 detach | Token-Ring-Schnittstelle entfernen |
| ifconfig at0 detach | ATM-Schnittstelle entfernen |

6. Geben Sie den entsprechenden Befehl **rmdev**, wie in den folgenden Beispielen gezeigt, ein, um den Adapter zu dekonfigurieren und dessen Einheitenkonfiguration in der Objektklasse für angepasste Einheiten zu *behalten*.

| Eintrag | Beschreibung |
|---------------|--|
| rmdev -l ent0 | Ethernet-Adapter dekonfigurieren |
| rmdev -l tok1 | Token-Ring-Adapter dekonfigurieren |
| rmdev -l atm1 | ATM-Adapter dekonfigurieren |
| rmdev -p pci1 | Die untergeordneten Einheiten eines PCI-Busses und alle deren untergeordnete Einheiten entfernen, aber die Einheitsdefinitionen in der Objektklasse für angepasste Einheiten behalten. |

Anmerkung: Zum Dekonfigurieren des Adapters und *Entfernen* der Einheitsdefinition aus der Objektklasse für angepasste Einheiten können Sie den Befehl **rmdev** mit dem Flag **-d** verwenden.

Achtung: Verwenden Sie das Flag **-d** *nur* dann mit dem Befehl **rmdev** für eine Hot-Plug-Operation, wenn Sie den Adapter entfernen (und nicht austauschen) möchten.

Zugehörige Tasks:

„ATM-Adapter dekonfigurieren“ auf Seite 591

Sie müssen alle emulierten LAN-Einheiten dekonfigurieren, bevor Sie den Adapter entfernen können.

WAN-Adapter dekonfigurieren:

Sie können einen WAN-Adapter dekonfigurieren.

Gehen Sie zum Dekonfigurieren eines WAN-Adapters wie folgt vor:

1. Geben Sie `lsslot -c pci` ein, um alle Hot-Plug-Steckplätze in der Systemeinheit aufzulisten und ihre Merkmale anzuzeigen.
2. Geben Sie den entsprechenden, in den folgenden Beispielen gezeigten SMIT-Befehl ein, um installierte Adapter aufzulisten und den aktuellen Status aller Einheiten in der Systemeinheit anzuzeigen:

| Eintrag | Beschreibung |
|------------------|---|
| smit 331121b9_ls | WAN-Multiprotokolladapter mit 2 Anschlüssen auflisten |
| smit riciophx_ls | WAN-Adapter vom Typ ARTIC auflisten |

Die folgende Namenskonvention wird für die unterschiedlichen Adaptertypen verwendet:

| Name | Adaptertyp |
|--------|---|
| dpmpa | Multiprotokolladapter mit 2-Anschlüssen |
| riciop | Adapter vom Typ ARTIC960 |

3. Geben Sie `lsdev -C -c Anschluss` ein, um die X.25-Anschlüsse an Ihrem Host aufzulisten. Es erscheint eine Nachricht wie die folgende:

```

sx25a0 Available 00-05-01-00 X.25 Port
x25s0 Available 00-05-01-00-00 V.3 X.25 Emulator

```

4. Schließen Sie alle Anwendungen, die den Adapter verwenden, den Sie dekonfigurieren möchten. Zum Fortsetzen dieser Prozedur müssen die Netzspeicherauszugsadressen auf dem System inaktiviert werden. Gehen Sie wie folgt vor, um die Netzspeicherauszugsadressen zu suchen und zu inaktivieren:
 - a. Geben Sie in einer Befehlszeile Folgendes ein:

smit dump

- b. Wählen Sie **Aktuelle Speicherauszugseinheiten anzeigen** aus.
 - c. Prüfen Sie, ob für eine der konfigurierten Speicherauszugseinheiten eine Netzadresse ausgewiesen wird. Wenn nicht, beenden Sie SMIT, und fahren Sie mit Schritt 5 fort. Wenn Sie die Adresse einer Speicherauszugseinheit in eine lokale Adresse ändern möchten, wählen Sie **Abbrechen** aus, oder drücken Sie die Taste F3, um mit dem nächsten Schritt fortzufahren.
 - d. Wenn für die primäre Speicherauszugseinheit eine Netzadresse angezeigt wird, ändern Sie sie in eine lokale Adresse. Wählen Sie dazu **Primäre Speicherauszugseinheit ändern** aus, und geben Sie anschließend im Feld **Primäre Speicherauszugseinheit** die lokale Adresse ein.
 - e. Wenn für die sekundäre Speicherauszugseinheit eine Netzadresse angezeigt wird, ändern Sie sie in eine lokale Adresse. Wählen Sie dazu **Sekundäre Speicherauszugseinheit ändern** aus, und geben Sie anschließend im Feld **Sekundäre Speicherauszugseinheit** die lokale Adresse ein.
 - f. Klicken Sie abschließend auf **OK**, oder drücken Sie die Eingabetaste.
5. Verwenden Sie die Befehle in der folgenden Tabelle, um die Einheitentreiber und Emulatoranschlüsse für diese Adapter zu dekonfigurieren und zu entfernen:

| Name | Multiprotokolladapter mit 2-Anschlüssen |
|-------------------|---|
| smit rmhdlcdpmpdd | Einheit dekonfigurieren |
| smit rmsdlcscied | SDLC-COMIO-Emulator dekonfigurieren |

| Name | PCI-Adapter vom Typ ARTIC960Hx |
|----------------|-------------------------------------|
| smit rmtsdd | Einheitentreiber dekonfigurieren |
| smit rmtsports | MPQP-COMIO-Emulationsport entfernen |

Adapter vom Typ IBM 4-Port 10/100 Base-TX Ethernet PCI dekonfigurieren:

Der Adapter IBM 4-Port 10/100 Base-TX Ethernet PCI hat vier Ethernet-Anschlüsse, und jeder dieser Anschlüsse muss dekonfiguriert werden, bevor Sie den Adapter entfernen können.

1. Geben Sie `lsslot -c pci` ein, um alle Hot-Plug-Steckplätze in der Systemeinheit aufzulisten und ihre Merkmale anzuzeigen.
2. Geben Sie `smit lsdenet` ein, um alle Einheiten in der Unterklasse PCI aufzulisten. Es erscheint eine Nachricht wie die folgende:

```
ent1 Available 1N-00 IBM 4-Port 10/100 Base-TX Ethernet PCI Adapter (23100020) (Port 1)
ent2 Available 1N-08 IBM 4-Port 10/100 Base-TX Ethernet PCI Adapter (23100020) (Port 2)
ent3 Available 1N-10 IBM 4-Port 10/100 Base-TX Ethernet PCI Adapter (23100020) (Port 3)
ent4 Available 1N-18 IBM 4-Port 10/100 Base-TX Ethernet PCI Adapter (23100020) (Port 4)
```
3. Schließen Sie alle Anwendungen, die den Adapter verwenden, den Sie dekonfigurieren möchten. Zum Fortsetzen dieser Prozedur müssen die Netzspeicherauszugsadressen auf dem System inaktiviert werden. Gehen Sie wie folgt vor, um die Netzspeicherauszugsadressen zu suchen und zu inaktivieren:
 - a. Geben Sie in einer Befehlszeile Folgendes ein:
smit dump
 - b. Wählen Sie **Aktuelle Speicherauszugseinheiten anzeigen** aus.
 - c. Prüfen Sie, ob für eine der konfigurierten Speicherauszugseinheiten eine Netzadresse ausgewiesen wird. Wenn nicht, beenden Sie SMIT und fahren mit Schritt 4 fort. Wenn Sie die Adresse einer Speicherauszugseinheit in eine lokale Adresse ändern möchten, wählen Sie **Abbrechen** aus, oder drücken Sie die Taste F3, um mit dem nächsten Schritt fortzufahren.
 - d. Wenn für die primäre Speicherauszugseinheit eine Netzadresse angezeigt wird, ändern Sie sie in eine lokale Adresse. Wählen Sie dazu **Primäre Speicherauszugseinheit ändern** aus, und geben Sie anschließend im Feld **Primäre Speicherauszugseinheit** die lokale Adresse ein.

- e. Wenn für die sekundäre Speicherauszugseinheit eine Netzadresse angezeigt wird, ändern Sie sie in eine lokale Adresse. Wählen Sie dazu **Sekundäre Speicherauszugseinheit ändern** aus, und geben Sie anschließend im Feld **Sekundäre Speicherauszugseinheit** die lokale Adresse ein.
 - f. Klicken Sie abschließend auf **OK**, oder drücken Sie die Eingabetaste.
4. Geben Sie `netstat -i` ein, um eine Liste aller konfigurierten Schnittstellen anzuzeigen und festzustellen, ob Ihr Adapter für TCP/IP konfiguriert ist. Es erscheint eine Ausgabe wie die folgende:

| Name | Mtu | Network | Address | Ipkts | Ierrs | Opkts | Oerrs | Coll |
|------|-------|----------|-----------------|-------|-------|-------|-------|------|
| lo0 | 16896 | link#1 | | 076 | 0 | 118 | 0 | 0 |
| lo0 | 16896 | 127 | 127.0.0.1 | 076 | 0 | 118 | 0 | 0 |
| lo0 | 16896 | :::1 | | 076 | 0 | 118 | 0 | 0 |
| tr0 | 1492 | link#2 | 8.0.5a.b8.b.ec | 151 | 0 | 405 | 11 | 0 |
| tr0 | 1492 | 19.13.97 | 19.13.97.106 | 151 | 0 | 405 | 11 | 0 |
| at0 | 9180 | link#3 | 0.4.ac.ad.e0.ad | 0 | 0 | 0 | 0 | 0 |
| at0 | 9180 | 6.6.6 | 6.6.6.5 | 0 | 0 | 0 | 0 | 0 |
| en0 | 1500 | link#5 | 0.11.0.66.11.1 | 212 | 0 | 1 | 0 | 0 |
| en0 | 1500 | 8.8.8 | 8.8.8.106 | 212 | 0 | 1 | 0 | 0 |

Ethernet-Adapter können zwei Schnittstellen haben, z. B. `et0` und `en0`.

- 5. Verwenden Sie den Befehl `ifconfig`, um jede Schnittstelle aus der Netzschnittstellenliste zu entfernen. Geben Sie beispielsweise `ifconfig en0 detach` ein, um die Standard-Ethernet-Schnittstelle zu entfernen, und geben Sie `ifconfig et0` ein, um die IEEE-802.3-Schnittstelle zu entfernen.
- 6. Verwenden Sie den Befehl `rmdev`, um den Adapter zu dekonfigurieren, aber die Einheitenkonfiguration in der Objektklasse für angepasste Einheiten zu behalten. Beispiel: `rmdev -l ent0`.

Anmerkung: Zum Dekonfigurieren des Adapters und *Entfernen* der Einheitsdefinition aus der Objektklasse für angepasste Einheiten können Sie den Befehl `rmdev` mit dem Flag `-d` verwenden.

Achtung: Verwenden Sie das Flag `-d` *nur* dann mit dem Befehl `rmdev` für eine Hot-Plug-Operation, wenn Sie den Adapter entfernen (und nicht austauschen) möchten.

ATM-Adapter dekonfigurieren:

Sie müssen alle emulierten LAN-Einheiten dekonfigurieren, bevor Sie den Adapter entfernen können.

Klassische IP- und LAN-Emulationsprotokolle können über ATM-Adapter ausgeführt werden. Das LAN-Emulationsprotokoll ermöglicht die Implementierung emulierter LANs über ein ATM-Netz. Emulierte LAN-Typen sind Ethernet/IEEE 802.3, Token-Ring/IEEE 802.5 und MPOA (MultiProtocol Over ATM).

Gehen Sie wie folgt vor, um eine LAN-Schnittstelle zu entfernen:

- 1. Geben Sie `lsslot -c pci` ein, um alle Hot-Plug-Steckplätze in der Systemeinheit aufzulisten und ihre Merkmale anzuzeigen.
- 2. Geben Sie mit `ls_atm` ein, um alle ATM-Adapter aufzulisten. Es erscheint eine Nachricht wie die folgende:

```
.
.
atm0 Available 04-04 IBM PCI 155 Mbps ATM Adapter (14107c00)
atm1 Available 04-06 IBM PCI 155 Mbps ATM Adapter (14104e00)
```

- 3. Geben Sie mit `listall_atmle` ein, um alle emulierten LAN-Clients auf den Adaptern aufzulisten. Es erscheint eine Nachricht wie die folgende:

```
ent1 Available ATM LAN Emulation Client (Ethernet)
ent2 Available ATM LAN Emulation Client (Ethernet)
ent3 Available ATM LAN Emulation Client (Ethernet)
tok1 Available ATM LAN Emulation Client (Token Ring)
tok2 Available ATM LAN Emulation Client (Token Ring)
```

Auf allen ATM-Adaptoren können mehrere emulierte Clients ausgeführt werden.

4. Geben Sie mit `listall_mpoa` ein, um alle emulierten LAN-Clients auf den Adaptern aufzulisten. Es erscheint eine Nachricht wie die folgende:

```
mpc0 Available      ATM LAN Emulation MPOA Client
```

`atm0` und `atm1` sind die physischen ATM-Adapter. `mpc0` ist ein mit MPOA emulierter Client. `ent1`, `ent2`, `ent3`, `tok1` und `tok2` sind im LAN emulierte Clients.

5. Geben Sie `entstat` ein, um festzustellen, auf welchem Adapter der Client ausgeführt wird. Es erscheint eine Nachricht wie die folgende:

```
-----  
ETHERNET STATISTICS (ent1) :  
Device Type: ATM LAN EmulationATM Hardware Address: 00:04:ac:ad:e0:ad  
.  
.  
.  
ATM LAN Emulation Specific Statistics:  
-----  
Emulated LAN Name: ETHelan3  
Local ATM Device Name: atm0  
Local LAN MAC Address:  
.  
.
```

6. Schließen Sie alle Anwendungen, die den Adapter verwenden, den Sie dekonfigurieren möchten. Zum Fortsetzen dieser Prozedur müssen die Netzspeicherauszugsadressen auf dem System inaktiviert werden. Gehen Sie wie folgt vor, um die Netzspeicherauszugsadressen zu suchen und zu inaktivieren:

- a. Geben Sie in einer Befehlszeile Folgendes ein:

```
smit dump
```

- b. Wählen Sie **Aktuelle Speicherauszugseinheiten anzeigen** aus.

- c. Prüfen Sie, ob für eine der konfigurierten Speicherauszugseinheiten eine Netzadresse ausgewiesen wird. Wenn nicht, beenden Sie SMIT und fahren mit Schritt 7 fort. Wenn Sie die Adresse einer Speicherauszugseinheit in eine lokale Adresse ändern möchten, wählen Sie **Abbrechen** aus, oder drücken Sie die Taste F3, um mit dem nächsten Schritt fortzufahren.

- d. Wenn für die primäre Speicherauszugseinheit eine Netzadresse angezeigt wird, ändern Sie sie in eine lokale Adresse. Wählen Sie dazu **Primäre Speicherauszugseinheit ändern** aus, und geben Sie anschließend im Feld **Primäre Speicherauszugseinheit** die lokale Adresse ein.

- e. Wenn für die sekundäre Speicherauszugseinheit eine Netzadresse angezeigt wird, ändern Sie sie in eine lokale Adresse. Wählen Sie dazu **Sekundäre Speicherauszugseinheit ändern** aus, und geben Sie anschließend im Feld **Sekundäre Speicherauszugseinheit** die lokale Adresse ein.

- f. Klicken Sie abschließend auf **OK**, oder drücken Sie die Eingabetaste.

7. Verwenden Sie den Befehl `rmdev -l Einheit`, um die Schnittstellen in der folgenden Reihenfolge zu dekonfigurieren:

- Emulierte Schnittstelle = `en1`, `et1`, `en2`, `et2`, `tr1`, `tr2` ...
- Emulierte Schnittstelle = `ent1`, `ent2`, `tok1`, `tok2` ...
- Multiprotocol Over ATM (MPOA) = `mpc0`
- ATM-Adapter = `atm0`

8. Geben Sie Folgendes ein, um den SCSI-Adapter `scsi1` und alle seine untergeordneten Einheiten zu dekonfigurieren, aber die Einheitendefinitionen in der Objektklasse für angepasste Einheiten zu behalten:

```
rmdev -R scsi1
```

Das System zeigt eine Nachricht wie die folgende an:

```
rmt0 Defined  
hdisk1 Defined  
scsi1 Defined
```


9. Geben Sie Folgendes ein, wenn Sie nur die untergeordneten Einheiten des SCSI-Adapters `scsi1`, den Adapter selbst aber nicht dekonfigurieren möchten und die Einheitendefinitionen in der Objektklasse für angepasste Einheiten beibehalten werden sollen:

```
rmdev -p scsi1
```

Das System zeigt eine Nachricht wie die folgende an:

```
rmt0 Defined
hdisk1 Defined
```

10. Geben Sie Folgendes ein, um die untergeordneten Einheiten des PCI-Busses `pci1` und deren untergeordnete Einheiten zu dekonfigurieren, aber die Einheitendefinitionen in der Objektklasse für angepasste Einheiten zu behalten:

```
rmdev -p pci1
```

Das System zeigt eine Nachricht wie die folgende an:

```
rmt0 Defined
hdisk1 Defined
scsi1 Defined
ent0 Defined
```

Zugehörige Tasks:

„Ethernet-, Token-Ring-, FDDI- und ATM-Adapter dekonfigurieren“ auf Seite 587

Führen Sie die folgenden Schritte aus, um einen Ethernet-, Token-Ring-, FDDI- oder ATM-Adapter zu dekonfigurieren:

Speicheradapter dekonfigurieren

Bevor Sie einen Speicheradapter entfernen oder austauschen können, müssen Sie diesen Adapter dekonfigurieren.

Zur Ausführung dieser Tasks müssen Sie als Root angemeldet sein.

Mit den folgenden Schritten dekonfigurieren Sie SCSI- und Fibre-Channel-Speicheradapter.

Die Dekonfiguration eines Speicheradapters umfasst die folgenden Tasks:

- Alle Anwendungen schließen, die den Adapter verwenden, den Sie entfernen, austauschen oder versetzen möchten
- Dateisysteme abhängen
- Sicherstellen, dass alle an den Adapter angeschlossenen Einheiten identifiziert und gestoppt werden
- Alle Steckplätze, die derzeit im Gebrauch sind, oder einen Steckplatz auflisten, in dem ein bestimmter Adapter installiert ist
- Steckplatzposition des Adapters ermitteln
- Übergeordnete und untergeordnete Einheiten als nicht verfügbar kennzeichnen
- Adapter als nicht verfügbar kennzeichnen

SAS-, SCSI-, und Fibre Channel-Adapter dekonfigurieren

Speicheradapter sind im Allgemeinen die übergeordneten Einheiten von Einheiten für austauschbare Speichermedien, wie z. B. Platten- oder Bandlaufwerke. Wenn Sie die übergeordnete Einheit entfernen, müssen alle angeschlossenen untergeordneten Einheiten entfernt oder in den Status "Defined" (Definiert) versetzt werden.

Gehen Sie zum Dekonfigurieren von SCSI- und Fibre-Channel-Adaptoren wie folgt vor:

1. Schließen Sie alle Anwendungen, die den Adapter verwenden, den Sie dekonfigurieren möchten.
2. Geben Sie `lsslot-c pci` ein, um alle Hot-Plug-Steckplätze in der Systemeinheit aufzulisten und ihre Merkmale anzuzeigen.

3. Geben Sie `lsdev -C` ein, um den aktuellen Status aller Einheiten in der Systemeinheit aufzulisten.
4. Geben Sie `umount` ein, um zuvor angehängte Dateisysteme, Verzeichnisse oder Dateien, die diesen Adapter verwenden, abzuhängen. Weitere Informationen finden Sie unter "JFS oder JFS2 anhängen".
5. Geben Sie `rmdev -l Adapter -R` ein, um den Adapter als nicht verfügbar zu kennzeichnen.

Achtung: Verwenden Sie das Flag `-d` für Hot-Plug-Operationen *nicht* mit dem Befehl `rmdev`, weil dies dazu führt, dass Ihre Konfiguration entfernt wird.

Zugehörige Konzepte:

„Hot-Plug-Verwaltung von PCI-Einheiten“ auf Seite 570

Sie können einen neuen PCI-Hot-Plug-Adapter in einem verfügbaren PCI-Steckplatz installieren, während das Betriebssystem aktiv ist.

Zugehörige Tasks:

„Einen PCI-Hot-Plug-Adapter entfernen“ auf Seite 573

Sie können einen PCI-Hot-Plug-Adapter aus der Systemeinheit entfernen, ohne das Betriebssystem herunterzufahren oder das System ausschalten zu müssen. Wenn Sie einen Adapter entfernen, stehen die von diesem Adapter bereitgestellten Ressourcen dem Betriebssystem und den Anwendungen nicht mehr zur Verfügung.

Asynchrone Adapter dekonfigurieren

Sie können einen asynchronen Adapter dekonfigurieren.

Zur Ausführung dieser Tasks müssen Sie als Root angemeldet sein.

Im Folgenden werden die Schritte für das Dekonfigurieren eines asynchronen Adapters beschrieben.

Bevor Sie einen asynchronen Adapter entfernen oder austauschen können, müssen Sie diesen Adapter dekonfigurieren. Die Dekonfiguration eines asynchronen Adapters umfasst die folgenden Tasks:

- Alle Anwendungen schließen, die den Adapter verwenden, den Sie entfernen, austauschen oder versetzen möchten
- Sicherstellen, dass alle an den Adapter angeschlossenen Einheiten identifiziert und gestoppt werden
- Alle Steckplätze, die derzeit im Gebrauch sind, oder einen Steckplatz auflisten, in dem ein bestimmter Adapter installiert ist
- Steckplatzposition des Adapters ermitteln
- Übergeordnete und untergeordnete Einheiten als nicht verfügbar kennzeichnen
- Adapter als nicht verfügbar kennzeichnen

Vorgehensweise

Bevor Sie einen asynchronen Adapter austauschen oder entfernen können, müssen Sie den Adapter und alle Einheiten, die von diesem Adapter gesteuert werden, dekonfigurieren. Zum Dekonfigurieren der Einheiten müssen Sie alle Prozesse beenden, die diese Einheiten verwenden. Führen Sie die folgenden Schritte aus:

1. Schließen Sie alle Anwendungen, die den Adapter verwenden, den Sie dekonfigurieren möchten.
2. Geben Sie `lsslot -c pci` ein, um alle Hot-Plug-Steckplätze in der Systemeinheit aufzulisten und ihre Merkmale anzuzeigen.
3. Geben Sie `lsdev -C -c tty` ein, um alle verfügbaren nicht grafikfähigen Terminals und den aktuellen Status aller Einheiten in der Systemeinheit aufzulisten.
4. Geben Sie `lsdev -C -c printer` ein, um alle Drucker- und Plottereinheiten aufzulisten, die an den Adapter angeschlossen sind.
5. Verwenden Sie den Befehl `rmdev`, um den Adapter als nicht verfügbar zu kennzeichnen.

Achtung: Verwenden Sie das Flag `-d` für Hot-Plug-Operationen *nicht* mit dem Befehl `rmdev`, weil dies dazu führt, dass Ihre Konfiguration entfernt wird.

Zugehörige Konzepte:

„Hot-Plug-Verwaltung von PCI-Einheiten“ auf Seite 570

Sie können einen neuen PCI-Hot-Plug-Adapter in einem verfügbaren PCI-Steckplatz installieren, während das Betriebssystem aktiv ist.

„Einheitenstatus“ auf Seite 564

Einheiten, die mit dem System verbunden sind, können vier Status annehmen.

Zugehörige Tasks:

„Einen PCI-Hot-Plug-Adapter entfernen“ auf Seite 573

Sie können einen PCI-Hot-Plug-Adapter aus der Systemeinheit entfernen, ohne das Betriebssystem herunterzufahren oder das System ausschalten zu müssen. Wenn Sie einen Adapter entfernen, stehen die von diesem Adapter bereitgestellten Ressourcen dem Betriebssystem und den Anwendungen nicht mehr zur Verfügung.

Zugehörige Informationen:

Printing administration

Fehlerbehebung bei E/A-Einheiten

Sie können die Ursache für Einheitenfehler bestimmen.

Einheitensoftware überprüfen

Sie können einen Fehler in der Einheitensoftware wie folgt korrigieren:

- Fehlerprotokoll überprüfen
- Alle Einheiten auflisten
- Status einer Einheit überprüfen
- Attribute einer Einheit überprüfen
- Attribute einer Einheit ändern
- Einheit mit einer anderen Anwendung verwenden
- Eine neue Einheit definieren

Fehlerprotokoll überprüfen

Stellen Sie fest, ob im Fehlerprotokoll Fehler für die Einheit, ihren Adapter oder die Anwendung, die die Einheit verwendet, protokolliert wurden. Informationen zur Durchführung dieser Überprüfung finden Sie im Abschnitt "Error Logging Facility". Kehren Sie nach Abschluss der Prozeduren zu diesem Schritt zurück.

Konnten Sie das Problem mit der Einheit beheben?

Wenn Sie das Problem mit der beschriebenen Methode nicht beheben können, fahren Sie mit dem nächsten Schritt, **Einheiten auflisten**, fort, um alle Einheiten aufzulisten.

Einheiten auflisten

Verwenden Sie den Befehl **lsdev -C**, um alle definierten oder verfügbaren Einheiten aufzulisten. Dieser Befehl zeigt die Merkmale aller Einheiten im System an.

Wenn die Einheit in der Einheitenliste aufgeführt ist, fahren Sie mit dem nächsten Schritt, **Einheitenstatus überprüfen**, fort, um den Status der Einheit zu überprüfen.

Wenn die Einheit nicht in der Einheitenliste aufgeführt ist, definieren Sie gemäß der Beschreibung im Schritt **Neue Einheitsdefinition** eine neue Einheit.

Einheitenstatus überprüfen

Suchen Sie die Einheit in der Liste, die vom Befehl **lsdev -C** generiert wurde. Prüfen Sie, ob die Einheit den Status `Available` (Verfügbar) hat.

Wenn die Einheit den Status `Available` hat, fahren Sie mit dem nächsten Schritt, **Einheitenattribute überprüfen**, fort.

Wenn die Einheit nicht den Status `Available` hat, definieren Sie gemäß der Beschreibung im Schritt **Neue Einheitsdefinition** eine neue Einheit.

Einheitenattribute überprüfen

Verwenden Sie den Befehl **lsattr -E -l *Einheitenname***, um die Attribute Ihrer Einheit aufzulisten.

Der Befehl **lsattr** zeigt Attributmerkmale und gültige Werte der Attribute für Einheiten im System an. Die korrekten Einstellungen können Sie der Dokumentation zur jeweiligen Einheit entnehmen.

Wenn die Einheitenattribute korrekt definiert sind, fahren Sie mit dem Abschnitt **Einheit mit einer anderen Anwendung verwenden** fort.

Wenn die Einheitenattribute nicht korrekt definiert sind, fahren Sie mit dem nächsten Schritt, **Einheitenattribute ändern**, fort.

Einheitenattribute ändern

Verwenden Sie den Befehl **chdev -l *Name* -a *Attribut=Wert***, um Einheitenattribute zu ändern. Bevor Sie diesen Befehl ausführen, sollten Sie den Abschnitt zu diesem Befehl in der Veröffentlichung *Commands Reference, Volume 1* lesen.

Der Befehl **chdev** ändert die Merkmale der Einheit, die Sie mit dem Flag **-l *Name*** angeben.

Wenn Sie das Problem durch das Ändern der Attribute nicht beheben können, fahren Sie mit dem nächsten Schritt, **Einheit mit einer anderen Anwendung verwenden**, fort.

Einheit mit einer anderen Anwendung verwenden

Versuchen Sie, die Einheit mit einer anderen Anwendung zu verwenden. Wenn die Einheit mit der anderen Anwendung ordnungsgemäß funktioniert, liegt das Problem möglicherweise bei der ersten Anwendung.

Wenn die Einheit ordnungsgemäß mit einer anderen Anwendung funktioniert, liegt das Problem möglicherweise bei der ersten Anwendung. Melden Sie den Fehler dem Softwarekundendienst.

Wenn die Einheit nicht ordnungsgemäß mit einer anderen Anwendung funktioniert, fahren Sie mit dem nächsten Schritt, **Neue Einheitsdefinition**, fort.

Neue Einheitsdefinition

Anmerkung: Sie müssen Rootberechtigung haben oder Mitglied der Sicherheitsgruppe sein, um den Befehl **mkdev** verwenden zu können.

Verwenden Sie den Befehl **mkdev**, um dem System eine Einheit hinzuzufügen.

Der Befehl **mkdev** kann eine neue Einheit definieren und verfügbar machen oder eine bereits definierte Einheit verfügbar machen. Sie können die vordefinierte Einheit mit jeder Kombination der Flags **-c**, **-s**

und **-t** eindeutig identifizieren. Bevor Sie diesen Befehl ausführen, sollten Sie den Abschnitt zu diesem Befehl in der Veröffentlichung *Commands Reference, Volume 3* lesen.

Wenn Sie das Problem durch die Definition der Einheit nicht beheben können, können Sie mit der Fehlerbestimmung aufhören und das Problem dem Kundendienst melden oder Ihre Einheit mit einem Diagnoseprogramm testen.

Einheitenanschlüsse überprüfen:

Führen Sie die folgenden Schritte aus, um Ihre Einheitenanschlüsse zu überprüfen:

1. Stellen Sie sicher, dass die Netzsteckdose mit Strom versorgt wird.
2. Stellen Sie sicher, dass das Netzkabel der Einheit ordnungsgemäß in die Einheit und in die Netzsteckdose eingesteckt ist.
3. Stellen Sie sicher, dass das Signalkabel ordnungsgemäß in die Einheit und in den richtigen Anschluss an der Systemeinheit eingesteckt ist.
4. Für SCSI-Einheiten stellen Sie sicher, dass der SCSI-Abschlussstecker ordnungsgemäß angebracht und die Einstellung der SCSI-Adresse korrekt ist.
5. Für Übertragungseinheiten stellen Sie sicher, dass die Einheit ordnungsgemäß an die Übertragungsleitung angeschlossen ist.
6. Stellen Sie sicher, dass die Einheit eingeschaltet ist.

Spezielle Prozeduren für die Verkabelung und Konfiguration sowie weitere Fehlerbehebungsinformationen können Sie der Dokumentation zu Ihrer Einheit entnehmen.

Fahren Sie mit dem nächsten Schritt fort, wenn das Problem nach der Ausführung der in diesem Abschnitt beschriebenen Schritte nicht behoben ist.

Fehlerbehebung beim Entfernen eines Adapters:

Sie können Fehlermeldungen empfangen, wenn die Einheit geöffnet ist und Sie den Befehl **rmdev** zum Dekonfigurieren eines Adapters verwenden.

Wenn der folgende Typ von Nachricht angezeigt wird, wenn Sie den Befehl **rmdev** zum Dekonfigurieren eines Adapters verwenden, ist die Einheit geöffnet, weil Anwendungen möglicherweise weiterhin versuchen, auf den Adapter zuzugreifen, den Sie entfernen oder austauschen möchten.

```
#rmdev -l ent0
Methodenfehler (/usr/lib/methods/ucfgent):
    0514-062
    Die angeforderte Funktion kann nicht ausgeführt werden, weil
    die angegebene Einheit aktiv ist.
```

Zur Behebung des Fehlers müssen Sie alle Anwendungen ermitteln, die den Adapter noch verwenden, und diese schließen. Zu diesen Anwendungen können die folgenden gehören:

- TCP/IP
- SNA
- OSI
- IPX/SPX
- Novell NetWare
- Streams
- Generic Data Link Control (GDLC)
 - IEEE Ethernet DLC
 - Token-Ring DLC
 - FDDI DLC

SNA-Anwendungen (Systems Network Architecture)

Im Folgenden sind einige SNA-Anwendungen aufgeführt, die den Adapter verwenden können:

- DB2
- TXSeries (CICS & Encina)
- DirectTalk
- MQSeries
- HCON
- ADSM

Streams-Anwendungen

Im Folgenden sind einige Streams-basierte Anwendungen aufgeführt, die den Adapter verwenden können:

- IPX/SPX
- Novell NetWare Version 4 und Novell NetWare Services 4.1
- Verbindungen und NetBios für dieses Betriebssystem

Anwendungen, die in WAN-Adaptoren ausgeführt werden

Im Folgenden sind einige Anwendungen aufgeführt, die den WAN-Adapter verwenden können:

- SDLC
- Bisync
- X.25
- ISDN
- QLLC for X.25

TCP/IP-Anwendungen

Die Zuordnung aller TCP/IP-Anwendungen, die die Schnittstellenschicht verwenden, kann mit dem Befehl **ifconfig** aufgehoben werden. Dieser Befehl bewirkt, dass die Anwendungen, die TCP/IP verwenden, das zulässige Zeitlimit überschreiten und Benutzer warnen, dass die Schnittstelle inaktiv ist. Nachdem Sie den Adapter hinzugefügt bzw. ausgetauscht und den Befehl **ifconfig** für die erneute Zuordnung der Schnittstelle ausgeführt haben, wird die Ausführung der Anwendungen fortgesetzt.

Bereitschaftsstatus einer Einheit überprüfen:

Sie können prüfen, ob eine Einheit bereit ist.

Gehen Sie wie folgt vor, um festzustellen, ob die Einheit bereit ist:

1. Überprüfen Sie, ob die Bereitschaftsanzeige der Einheit leuchtet.
2. Überprüfen Sie, ob austauschbare Datenträger, wie z. B. Bänder, Disketten oder optische Einheiten, korrekt eingelegt sind.
3. Überprüfen Sie das Farbband, den Papier- und Tonerbestand für Drucker und Plotter.
4. Überprüfen Sie, ob der Datenträger Schreibzugriffe zulässt, wenn Sie versuchen, auf die Einheit zu schreiben.

Konnten Sie das Problem mit der Einheit durch Ihre Überprüfungen beheben? Wenn Sie das Problem durch die Überprüfung des Bereitschaftsstatus der Einheit nicht beheben können, fahren Sie mit dem nächsten Schritt fort.

Einheitendiagnose:

Führen Sie Ihre Hardwarediagnose aus, um festzustellen, ob eine Einheit defekt ist.

Wenn bei der Hardwarediagnose kein Problem mit der Einheit festgestellt wird, überprüfen Sie die Einheitensoftware. Wenn Ihre Einheit die Diagnosetests besteht, liegt möglicherweise ein Problem in der Zusammenarbeit zwischen Einheit und Systemsoftware vor. In einem solchen Fall müssen Sie das Problem Ihrer Softwareserviceorganisation melden.

Zieleinheitenkonfiguration

Der Befehl `cfgmgr` kann für einen begrenzten Bereich von Zielkonfigurationen für E/A-Einheiten mit dem Flag `-c` als Verbindungsoption verwendet werden.

Zugehörige Informationen:

`cfgmgr` command

Zielkonfiguration von FC- und FCoE-Einheiten

Die Option `cfgmgr -c` wird mit FC- (Fibre Channel) und FCoE-Adaptoren (Fibre Channel over Ethernet) für Zielkonfigurationen verwendet.

Der Befehl `cfgmgr` kann für einen begrenzten Bereich von Einheitenkonfigurationen mit dem Flag `-c` als Verbindungsoption verwendet werden. Für FC- und FCoE-Adapter ist die Syntax folgende:

```
cfgmgr -l fscsi0 -c "Parameter=val[,Parameter=val,...]"
```

Wenn Sie die Verbindungsfilterzeichenfolge verwenden, können Sie den Bereich der Einheitenerkennung mit einem oder mehreren der folgenden Parametern begrenzen:

Tabelle 70. Parameter für das Flag `cfgmgr -c`

| Parametername | Beschreibung |
|------------------------|--|
| <code>ww_name</code> | Weltweiter Portname der Zieleinheit |
| <code>node_name</code> | Weltweiter Knotenname der Zieleinheit |
| <code>scsi_id</code> | Knotenport-ID der Zieleinheit, die der SCSI-ID (Small Computer System Interface) für FCP-Speichereinheiten (Fibre Channel Protocol) zugeordnet ist |
| <code>lun_id</code> | Nummer der logischen Einheit (LUN, Logical Unit Number) |

Der folgende Befehl beispielsweise konfiguriert eine einzelne LUN (`lun_id 0x100000000000`) am Speicherzielport, der den weltweiten Portnamen `0x5001738000330191` hat:

```
# cfgmgr -l fscsi0 -c "ww_name=0x5001738000330191,lun_id=0x100000000000"
```

Dieser Scan findet nur für den Hostadapterport `fscsi0` statt.

Anmerkungen:

- Die führenden Zeichen `0x` im Parameterwert sind optional.
- Alle Parameter müssen als Hexadezimalzahlen dargestellt werden.

Im folgenden Beispiel ist nur ein einziger Parameter angegeben:

```
# cfgmgr -l fscsi0 -c "lun_id=0x100000000000"
```

Dieser Befehl scannt alle Speichereinheitenports im Speicherbereichsnetz (SAN, Storage Area Network) und konfiguriert diese einzelne logische Einheit für jeden SAN-Zielport, an dem diese LUN vorhanden ist.

Richtlinien und Regeln für Verbindungsfilterparameter

Beachten Sie bei der Verwendung der Verbindungsfilterparameter die folgenden Punkte:

- Die Zielkonfiguration für FC- und FCoE-Einheiten gilt nur für Umgebungen mit Switch-Port. Wenn Sie eine Verbindungszeichenfolge angeben, die einem Zielport direkt zugeordnet ist, schlägt die Verbindung mit einer Nachricht fehl, in der Sie darauf hingewiesen werden, dass die untergeordneten Einheiten nicht gefunden wurden.
- Das Flag **-c** wird nur unterstützt, wenn es zusammen mit dem Flag **-l** des Befehls **cfgmgr** angegeben wird, das den Geltungsbereich des Befehls auf eine jeweils eine einzige `fscsiX`-Einheit beschränkt.
- Wenn Sie **-?** als Verbindungszeichenfolge für das Flag **-c** des Befehls **cfgmgr** zusammen mit dem Flag **-v** angeben, werden die Syntaxinformationen angezeigt.
- Wenn Sie doppelte Parameter (z. B. **lun_id** zweimal) angeben, tritt ein Fehler auf. Es werden keine Einheiten erkannt.
- Mit Ausnahme von Duplikaten ist jede Kombination der Parameter **lun_id**, **scsi_id**, **ww_name** und **node_name** zulässig. Zur eindeutigen Identifikation einer zu konfigurierenden LUN, eines zu konfigurierenden Ziels oder eines zu konfigurierenden Speicherknosens müssen Sie mindestens einen, vorzugsweise aber zwei Parameter angeben, aber es sind auch mehr Parameter zulässig. Die folgende Liste enthält den Parameter bzw. die Parameterkombination, die für eine eindeutige Identifizierung einer LUN, eines Ziels oder eines Speicherknosens erforderlich ist.
 - Die Parameter **ww_name** und **lun_id** identifizieren eine zu konfigurierende LUN an einem Zielport eindeutig.
 - Die Parameter **scsi_id** und **lun_id** identifizieren eine zu konfigurierende LUN an einem Zielport eindeutig.
 - Die Parameter **node_name** und **lun_id** konfigurieren eine LUN für alle Zielports für einen bestimmten Speicherknosens. Diese Parameter können die Zielports nur konfigurieren, wenn alle Zielports denselben Parameter **node_name** haben, was für einige Speichereinheiten zutreffen kann.
 - Der Parameter **ww_name** konfiguriert alle LUNs für ein bestimmtes Ziel.
 - Der Parameter **node_name** konfiguriert alle Zielports für einen bestimmten Speicherknosens (nur, wenn alle Zielports denselben Parameter **node_name** haben, was für einige Speichereinheiten zutreffen kann).
 - Der Parameter **lun_id** konfiguriert eine LUN an allen Zielports, die von dieser `fscsi`-Einheit aus sichtbar sind.
- Wenn mehr als zwei Parameter angegeben werden, verwendet der Konfigurationscode diese zusätzlichen Informationen, um die Einheitenposition zu validieren. Falls einer der angegebenen Parameterwert mit den im Speicherbereichsnetz gemeldeten Werten in Konflikt steht, schlägt der Befehl fehl, und es werden keine Einheiten konfiguriert.

Zugehörige Informationen:

`cfgmgr` command

Bandlaufwerke

Die hier beschriebenen Systemverwaltungsfunktionen beziehen sich auf Bandlaufwerke.

Viele dieser Funktionen ändern oder rufen Informationen aus der Einheitenkonfigurationsdatenbank ab, die Informationen zu den Einheiten in Ihrem System enthält. Die Einheitenkonfigurationsdatenbank setzt sich aus der vordefinierten Konfigurationsdatenbank, die Informationen zu allen möglichen Typen von Einheiten enthält, die im System unterstützt werden, und der angepassten Konfigurationsdatenbank zusammen, die Informationen zu den jeweiligen Einheiten enthält, die derzeit im System vorhanden sind. Damit das Betriebssystem ein Bandlaufwerk oder eine andere Einheit nutzen kann, muss die Einheit in der angepassten Konfigurationsdatenbank konfiguriert und ein Einheitentyp in der vordefinierten Konfigurationsdatenbank definiert sein.

Attribute für Bandlaufwerke

Sie können diese Attribute für Bandlaufwerke an die Anforderungen Ihres Systems anpassen.

Die Attribute können mit SMIT oder Befehlen (insbesondere den Befehlen **lsattr** und **chdev**) angezeigt oder geändert werden.

Jeder Bandlaufwerktyp verwendet nur eine Teilmenge der Attribute.

Allgemeine Informationen zu jedem Attribut

Blockgröße

Dieses Attribut gibt die Blockgröße an, die beim Lesen oder Beschreiben des Bandes verwendet wird. Die Daten werden in Datenblöcken auf das Band geschrieben, wobei zwischen den Blöcken jeweils ein Abstand eingehalten wird. Größere Datensätze sind hilfreich, wenn auf ein unformatiertes Band geschrieben wird, weil die Anzahl der Abstände zwischen den Datensätzen auf dem Band dadurch geringer wird und mehr Daten geschrieben werden können. Der Wert **0** steht für eine variable Blockgröße. Die zulässigen Werte und Standardwerte variieren je nach Bandlaufwerk.

Einheitenpuffer

Wenn Sie dieses Attribut (mit dem Befehl **chdev**) auf **mode=yes** setzen, wird eine Anwendung über die Beendigung des Schreibvorgangs benachrichtigt, nachdem die Daten in den Datenpuffer des Bandlaufwerks übertragen wurden, aber nicht zwingenderweise, nachdem die Daten tatsächlich auf das Band geschrieben wurden. Wenn Sie **mode=no** angeben, wird eine Anwendung nach Abschluss eines Schreibvorgangs erst benachrichtigt, nachdem die Daten tatsächlich auf das Band geschrieben wurden. Der Datenstrommodus kann für Lese- und Schreibvorgänge nicht aufrecht erhalten werden, wenn dieses Attribut den Wert **mode=no** hat. Der Standardwert ist **mode=yes**.

Mit dem Wert **mode=no** ist das Bandlaufwerk zwar langsamer, hat aber mehr vollständige Daten, falls der Strom oder das System ausfällt, und ermöglicht einen besseren Umgang mit EOM-Bedingungen (End of Media, Ende des Datenträgers).

Erweiterte Dateimarkierungen

Wenn Sie dieses Attribut (für den Befehl **chdev** das Attribut **extfm**) auf den Wert **no** setzen, wird eine reguläre Dateimarkierung auf das Band geschrieben, wenn eine Dateimarkierung geschrieben wird. Hat das Attribut den Wert **yes**, wird eine erweiterte Dateimarkierung geschrieben. Für Bandlaufwerke kann dieses Attribut aktiviert werden. Der Standardwert ist **no**. Erweiterte Dateimarkierungen auf 8-mm-Bandlaufwerken belegen beispielsweise 2,2 MB des Bands. Das Schreiben dieser Markierungen kann bis zu 8,5 Sekunden dauern. Reguläre Dateimarkierungen belegen 184 KB, und der Schreibvorgang dauert ungefähr 1,5 Sekunden.

Wenn Sie ein 8-mm-Band im Anfügemodus verwenden und Fehler reduzieren möchten, können Sie erweiterte Dateimarkierungen verwenden, um eine bessere Positionierung nach dem Zurückspulen an Dateimarkierungen zu erreichen.

Neu spannen

Wenn Sie das Attribut *Neu spannen* (für den Befehl **chdev** das Attribut **ret**) auf **ret=yes** setzen, wird das Bandlaufwerk angewiesen, ein Band automatisch erneut zu spannen, wenn ein Band eingelegt oder das Laufwerk zurückgesetzt wird. *Neu spannen* bedeutet, dass ein Band bis zum Ende vorgespult und anschließend an den Anfang zurückgespult wird, um eine gleichmäßige Spannung über das gesamte Band zu erzielen. Durch das *erneute Spannen* des Bands können Fehler reduziert werden, aber diese Aktion kann mehrere Minuten dauern. Wenn Sie den Wert **ret=no** angeben, spannt das Bandlaufwerk das Band nicht automatisch neu. Der Standardwert ist **yes**.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Die Einstellung für Schreibdichte #1 (für den Befehl **chdev** das Attribut **density_set_1**) setzt den Wert für die Dichte, mit dem das Bandlaufwerk schreibt, wenn die Gerätedateien `/dev/rmt*`,

/dev/rmt*.1, /dev/rmt*.2 und /dev/rmt*.3 verwendet werden. Die Einstellung für die Schreibdichte #2 (für den Befehl **chdev** das Attribut **density_set_2**) setzt den Wert für die Dichte, mit dem das Bandlaufwerk schreibt, wenn die Gerätedateien /dev/rmt*.4, /dev/rmt*.5, /dev/rmt*.6 und /dev/rmt*.7 verwendet werden. Weitere Informationen finden Sie im Abschnitt „Gerätedateien für Bandlaufwerke“ auf Seite 612.

Die Dichteeinstellungen werden als Dezimalzahlen aus dem Bereich 0 bis 255 angegeben. Wenn Sie null (0) einstellen, wird die Standarddichte für das Bandlaufwerk ausgewählt. Gewöhnlich ist dies die höchste Dichteeinstellung, die das Laufwerk unterstützt. Bestimmte zulässige Werte und ihre Bedeutungen variieren je nach Bandlaufwerktyp. Diese Attribute haben keine Auswirkung auf die Fähigkeit des Bandlaufwerks, Bänder zu lesen, die mit einer vom Bandlaufwerk unterstützten Dichte beschrieben wurden. Gewöhnlich wird die Einstellung für die Schreibdichte #1 auf die höchste vom Bandlaufwerk unterstützte Dichte gesetzt, und die Einstellung für Schreibdichte #2 auf den zweithöchsten Wert.

Unterstützung für Reservierung

Wenn Sie für Bandlaufwerke, die das Reservierungsattribut (für den Befehl **chdev** das Attribut **res_support**) verwenden, den Wert **res_support=yes** angeben, wird das Bandlaufwerk am SCSI-Bus reserviert, solange es geöffnet ist. Wenn mehrere SCSI-Adapter die Bändeinheit gemeinsam nutzen, wird damit gewährleistet, dass jeweils nur ein Adapter auf die Einheit zugreift, während sie geöffnet ist. Einige SCSI-Bandlaufwerke unterstützen die Reservierungs- und Freigabebefehle nicht. Einige SCSI-Bandlaufwerke haben einen vordefinierten Wert für dieses Attribut, so dass Reservierungs- und Freigabebefehle immer unterstützt werden.

Variable Blockgröße

Das Attribut für variable Blockgröße (für den Befehl **chdev** das Attribut **var_block_size**) gibt die Blockgröße an, die das Bandlaufwerk für das Schreiben von Sätzen variabler Länge voraussetzt. Einige SCSI-Bandlaufwerke erfordern, dass selbst dann eine Blockgröße ungleich null in ihrem Datensatz für Modusauswahl angegeben wird, wenn Sätze variabler Länge geschrieben werden. Das Attribut **Blockgröße** wird für Datensätze variabler Länge auf 0 gesetzt. Lesen Sie in der SCSI-Spezifikation für Ihr Bandlaufwerk nach, ob dies erforderlich ist oder nicht.

Datenkomprimierung

Wenn Sie das Attribut für Datenkomprimierung (für den Befehl **chdev** das Attribut **compress**) auf **compress=yes** setzen, wird das Bandlaufwerk im Komprimierungsmodus betrieben, sofern das Laufwerk in der Lage ist, Daten zu komprimieren. Wenn ja, schreibt das Laufwerk die Daten in komprimiertem Format auf das Band, so dass mehr Daten auf ein Band passen. Hat dieses Attribut den Wert **no**, wird das Bandlaufwerk gezwungen, im nativen Modus (ohne Komprimierung) auf das Band zu schreiben. Leseoperationen sind von der Einstellung dieses Attributs nicht betroffen. Die Standardeinstellung ist **yes**.

Autoloader

Wenn Sie das Attribut für Autoloader (für den Befehl **chdev** das Attribut **autoload**) auf **autoload=yes** setzen, wird der Autoloader aktiviert, sofern das Laufwerk entsprechend ausgestattet ist. Sollte dies der Fall und sich ein anderes Band im Loader befinden, wird jede Lese- oder Schreiboperation, die bis zum Ende des Bands geht, automatisch auf dem nächsten Band fortgesetzt. Die Bandlaufwerkbefehle, die auf eine Bandkassette beschränkt sind, sind nicht betroffen. Die Standardeinstellung ist **yes**.

Wiederholungsintervall

Das Attribut für das Wiederholungsintervall legt die Anzahl der Sekunden fest, die das System nach dem Scheitern eines Befehls wartet, bevor es den Befehl erneut absetzt. Das System kann einen gescheiterten Befehl bis zu vier Mal erneut absetzen. Dieses Attribut gilt nur für Bandlaufwerke des Typs OST. Die Standardeinstellung ist 45.

Zeitlimit für Lese-/Schreiboperationen

Das Attribut für das Zeitlimit für Lese-/Schreiboperationen bzw. die maximale Verzögerung für Lese-/Schreiboperationen (**READ/WRITE**) legt die maximale Anzahl an Sekunden fest, die das

System für die Ausführung eines Lese- oder Schreibbefehls zulässt. Dieses Attribut gilt nur für Bandlaufwerke des Typs OST. Die Standardeinstellung ist 144.

Fehler bei Bandwechsel zurückgeben

Dieses Attribut bewirkt (sofern gesetzt), dass ein Fehler zurückgegeben wird, wenn das Bandlaufwerk zurückgesetzt oder das Band gewechselt wird. Es muss eine vorherige Operation auf dem Bandlaufwerk stattgefunden haben, die das Band beim Schließen an einer anderen Position als dem Bandanfang belassen hat. Es wird der Fehler -1 zurückgegeben, und errno ist auf E10 gesetzt. Sobald die Anwendung über die Fehlerbedingung informiert ist, wird der Fehler behoben. Auch durch eine Rekonfiguration des Bandlaufwerks kann die Fehlerbedingung beseitigt werden.

Attribute für 4-mm-Bandlaufwerke mit 2,0 GB (Typ 4mm2gb):

Im Folgenden sind die Attribute für 4-mm-Bandlaufwerke mit 2,0 GB (Typ 4mm2gb) beschrieben.

Blockgröße

Der Standardwert ist 1024.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 4-mm-Bandlaufwerk mit 2,0 GB konfiguriert ist, haben die Attribute Neu spannen, Unterstützung für Reservierung, Variable Blockgröße, Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2 vordefinierte Werte, die nicht geändert werden können. Die Einstellungen für die Schreibdichte sind vordefiniert, weil das Bandlaufwerk für Schreiboperationen immer den 2,0-GB-Modus verwendet.

Attribute für 4-mm-Bandlaufwerke mit 4,0 GB (Typ 4mm4gb):

Im Folgenden sind die Attribute für 4-mm-Bandlaufwerke mit 4,0 GB (Typ 4mm4gb) beschrieben.

Blockgröße

Der Standardwert ist 1024.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Der Benutzer kann die Einstellung für Schreibdichte dieses Laufwerks nicht ändern. Die Einheit rekonfiguriert sich wie folgt automatisch für den jeweils installierten DDS-Datenträgertyp (Digital Data Storage):

| Datenträgertyp | Einheitenkonfiguration |
|-----------------------|---|
| DDS | Schreibgeschützt |
| DDS | Lese-/Schreiboperationen nur im 2,0-GB-Modus |
| DDS2 | Leseoperationen mit jeder Schreibdichte, Schreiboperationen nur im 4,0-GB-Modus |
| non-DDS | Nicht unterstützt. Die Kassette wird ausgeworfen. |

Datenkomprimierung

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 4-mm-Bandlaufwerk mit 4,0 GB konfiguriert ist, haben die Attribute Neu spannen, Unterstützung für Reservierung, Variable Blockgröße, Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2 vordefinierte Werte, die nicht geändert werden können.

Attribute für 8-mm-Bandlaufwerke mit 2,3 GB (Typ 8mm):

Im Folgenden sind die Attribute für 8-mm-Bandlaufwerke mit 2,3 GB (Typ 8mm) beschrieben.

Blockgröße

Der Standardwert ist 1024. Ein kleinerer Wert reduziert die Menge der Daten, die auf einem Band gespeichert werden.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 8-mm-Bandlaufwerk mit 2,3 GB konfiguriert ist, haben die Attribute Neu spannen, Unterstützung für Reservierung, Variable Blockgröße, Datenkomprimierung, Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2 vordefinierte Werte, die nicht geändert werden können. Die Einstellungen für die Schreibdichte sind vordefiniert, weil das Bandlaufwerk für Schreiboperationen immer den 2,3-GB-Modus verwendet.

Attribute für 8-mm-Bandlaufwerke mit 5,0 GB (Typ 8mm5gb):

Im Folgenden sind die Attribute für 8-mm-Bandlaufwerke mit 5,0 GB (Typ 8mm5gb) beschrieben.

Blockgröße

Der Standardwert ist 1024. Wenn ein Band im 2,3-GB-Modus beschrieben wird, reduziert ein kleinerer Wert die Menge der Daten, die auf einem Band gespeichert werden.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Die folgenden Einstellungen sind gültig:

| Einstellung | Bedeutung |
|--------------------|------------------------------------|
| 140 | 5-GB-Modus (Komprimierung möglich) |
| 21 | 5-GB-Modus (ohne Komprimierung) |
| 20 | 2,3-GB-Modus |
| 0 | Standardeinstellung (5,0-GB-Modus) |

Die Standardwerte sind 140 für Einstellung für Schreibdichte #1 und 20 für Einstellung für Schreibdichte #2. Wenn für die Einstellung für Schreibdichte #1 oder #2 der Wert 21 angegeben ist, kann der Benutzer Schreiboperationen auf einem nicht komprimierten Band im 5-GB-Modus durchführen.

Datenkomprimierung

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 8-mm-Bandlaufwerk mit 5,0 GB konfiguriert ist, haben die Attribute Neu spannen, Unterstützung für Reservierung und Variable Blockgröße vordefinierte Werte, die nicht geändert werden können.

Attribute für 8-mm-Bandlaufwerke mit 20000 MB (selbst-konfigurierend):

Im Folgenden sind die Attribute für 8-mm-Bandlaufwerke mit 20000 MB (selbst-konfigurierend) beschrieben.

Blockgröße

Der Standardwert ist 1024.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Das Laufwerk kann Datenkassetten im 20,0-GB-Format lesen und beschreiben. Bei einem Lesebefehl bestimmt das Laufwerk automatisch das Format auf dem Band. Bei einem Schreibbefehl bestimmt die Einstellung für Schreibdichte, in welchem Format die Daten auf das Band geschrieben werden.

Die folgenden Einstellungen sind gültig:

| Einstellung | Bedeutung |
|-------------|-------------------------------------|
| 39 | 20-GB-Modus (Komprimierung möglich) |
| 0 | Standardeinstellung (20,0-GB-Modus) |

Der Standardwert ist 39 für die Einstellungen für Schreibdichte #1 und #2.

Datenkomprimierung

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 8-mm-Bandlaufwerk mit 20,0 GB konfiguriert ist, haben die Attribute Neu spannen, Unterstützung für Reservierung und Variable Blockgröße vordefinierte Werte, die nicht geändert werden können.

Attribute für Bandlaufwerke mit 35 GB (Typ 35gb):

Im Folgenden sind die Attribute für Bandlaufwerke mit 35 GB (Typ 35gb) beschrieben.

Blockgröße

Der Durchsatz des Bandlaufwerks IBM 7205 Modell 311 ist maßgeblich von der Blockgröße abhängig. Die empfohlene Mindestblockgröße für dieses Laufwerk sind 32 KB. Jede Blockgröße kleiner als 32 KB wirkt sich negativ auf die Übertragungsgeschwindigkeit (Sicherungs-/Wiederherstellungszeit) aus. In der folgenden Liste sind die empfohlenen Blockgrößen nach Befehl aufgelistet:

| Unterstützter Befehl | Standardblockgröße (Bytes) | EMPFEHLUNG |
|----------------------|--|---|
| BACKUP | 32 K oder 51,2 K (Standardeinstellung) | In Abhängigkeit davon, ob die Sicherung nach Namen durchgeführt wird, werden entweder 32 K oder 51,2 K verwendet. Es ist keine Änderung erforderlich. |
| TAR | 10 K | Die im Handbuch angegebene Blockgröße von 512 KB ist falsch. Setzen Sie den Blockungsparameter auf -N64 . |
| MKSYSB | Siehe BACKUP . | Der Befehl MKSYSB verwendet den Befehl BACKUP . Es ist keine Änderung erforderlich. |
| DD | Nicht zutreffend | Setzen Sie den Parameter Blocking auf bs=32K . |
| CPIO | Nicht zutreffend | Setzen Sie den Parameter Blocking auf -C64 . |

Anmerkung: Berücksichtigen Sie bei der Auswahl einer Blockgröße die Kapazität und den Durchsatz. Geringe Blockgrößen können einen erheblichen Einfluss auf die Leistung und einen minimalen Einfluss auf die Kapazität haben. Wenn Sie kleinere Blockgrößen als die empfohlenen verwenden, wirkt sich dies auf die Kapazität des 2,6-GB-Formats (Schreibdichte) und des 6,0-GB-Formats (Schreibdichte) aus. Wenn Sie beispielsweise eine Blockgröße von 1024 Bytes für die Sicherung von 32 GB Daten verwenden, dauert der Vorgang ungefähr 22 Stunden. Die Sicherung derselben 32 GB Daten mit einer Blockgröße von 32 KB dauert ungefähr 2 Stunden.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Die folgende Tabelle zeigt die unterstützten Datenkassettentypen und Einstellungen für die Schreibdichte (in Dezimal- und Hexadezimalschreibweise) für das Bandlaufwerk IBM 7205-311. Wenn Sie eine Wiederherstellungsoperation (Lesen) durchführen, stellt das Bandlaufwerk die Schreibdichte automatisch auf die beim Schreiben gewählte Schreibdichte ein. Wenn Sie eine Sicherungsoperation (Schreiben) durchführen, müssen Sie die Einstellung für die Schreibdichte wählen, die für die verwendete Datenkassette richtig ist.

| Unterstützte Datenkassetten | Native Kapazität | Datenkapazität bei Komprimierung | SMIT-Einstellung für Schreibdichte | Einstellung für Schreibdichte in Hexadezimalschreibweise |
|-----------------------------|------------------|--|------------------------------------|--|
| DLTtape III | 2,6 GB | 2,6 GB (ohne Komprimierung) | 23 | 17h |
| | 6,0 GB | 6,0 GB (ohne Komprimierung) | 24 | 18h |
| | 10,0 GB | 20,0 GB (Standardeinstellung für Laufwerk) | 25 | 19h |
| DLTtapeIIIxt | 15,0 GB | 30,6 GB (Standardeinstellung für Laufwerk) | 25 | 19h |
| DLTtapeIV | 20,0 GB | 40,0 GB | 26 | 1Ah |
| | 35,0 GB | 70,0 GB (Standardeinstellung für Laufwerk) | 27 | 1Bh |

Anmerkung: Wenn Sie eine nicht unterstützte native Kapazität für die Datenkapazität anfordern, verwendet das Laufwerk standardmäßig die höchste unterstützte Kapazität für die Datenkassette, die in das Laufwerk eingelegt ist.

Datenkomprimierung

Die tatsächliche Komprimierung richtet sich nach dem Typ der geschriebenen Daten (siehe Tabelle oben). Für die Datenkapazität bei Komprimierung wird ein Komprimierungsverhältnis von 2:1 angenommen.

Attribute mit festen Werten

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Attribute für 1/4-Zoll-Bandlaufwerke mit 150 MB (Typ 150mb):

Im Folgenden sind die Attribute für 1/4-Zoll-Bandlaufwerke mit 150 MB (Typ 150mb) beschrieben.

Blockgröße

Die Standardblockgröße ist 512. Die einzige weitere gültige Blockgröße ist 0 für Blöcke variabler Länge.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Eine Schreiboperation auf ein 1/4-Zoll-Band ist nur am Bandanfang (BOT, Beginning of Tape) oder nach dem Erkennen eines leeren Bandes möglich. Wenn das Band Daten enthält, können Sie diese Daten nur überschreiben, wenn sich der Schreibkopf am Bandanfang befindet. Wenn Sie einem Band Daten hinzufügen möchten, das beschrieben und anschließend zurückgespult wurde, müssen Sie bis zur nächsten Dateimarkierung vorspulen, woraufhin das System einen Fehler zurückgibt. Erst dann können Sie erneut mit dem Schreiben beginnen.

Neu spannen

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Die folgenden Einstellungen sind gültig:

| Einstellung | Bedeutung |
|-------------|--|
| 16 | QIC-150 |
| 15 | QIC-120 |
| 0 | Standardeinstellung (QIC-150) bzw. die letzte Einstellung für die Schreibdichte, die von einem System verwendet wurde. |

Die Standardwerte sind 16 für Einstellung für Schreibdichte #1 und 15 für Einstellung für Schreibdichte #2.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 1/4-Zoll-Bandlaufwerk mit 150 MB konfiguriert ist, haben die Attribute Erweiterte Dateimarkierungen, Unterstützung für Reservierung, Variable Blockgröße und Datenkomprimierung vordefinierte Werte, die nicht geändert werden können.

Attribute für 1/4-Zoll-Bandlaufwerke mit 525 MB (Typ 525mb):

Im Folgenden sind die Attribute für 1/4-Zoll-Bandlaufwerke mit 525 MB (Typ 525mb) beschrieben.

Blockgröße

Die Standardblockgröße ist 512. Die anderen gültigen Blockgrößen sind 0 für Blöcke variabler Länge und 1024.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Eine Schreiboperation auf ein 1/4-Zoll-Band ist nur am Bandanfang (BOT, Beginning of Tape) oder nach dem Erkennen eines leeren Bandes möglich. Wenn das Band Daten enthält, können Sie diese Daten nur überschreiben, wenn sich der Schreibkopf am Bandanfang befindet. Wenn Sie einem Band Daten hinzufügen möchten, das beschrieben und anschließend zurückgespult wurde, müssen Sie bis zur nächsten Dateimarkierung vorspulen, woraufhin das System einen Fehler zurückgibt. Erst dann können Sie erneut mit dem Schreiben beginnen.

Neu spannen

Eine Schreiboperation auf ein 1/4-Zoll-Band ist nur am Bandanfang (BOT, Beginning of Tape) oder nach dem Erkennen eines leeren Bands möglich. Wenn das Band Daten enthält, können Sie diese Daten nur überschreiben, wenn sich der Schreibkopf am Bandanfang befindet. Wenn Sie einem Band Daten hinzufügen möchten, das beschrieben und anschließend zurückgespult wurde, müssen Sie bis zur nächsten Dateimarkierung vorspulen, woraufhin das System einen Fehler zurückgibt. Erst dann können Sie erneut mit dem Schreiben beginnen.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Die folgenden Einstellungen sind gültig:

| Einstellung | Bedeutung |
|-------------|--|
| 17 | QIC-525* |
| 16 | QIC-150 |
| 15 | QIC-120 |
| 0 | Standardeinstellung (QIC-525) bzw. die letzte Einstellung für die Schreibdichte, die von einem System verwendet wurde. |

* QIC-525 ist der einzige Modus, der die Blockgröße 1024 unterstützt.

Die Standardwerte sind 17 für Einstellung für Schreibdichte #1 und 16 für Einstellung für Schreibdichte #2.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 1/4-Zoll-Bandlaufwerk mit 525 MB konfiguriert ist, haben die Attribute Erweiterte Dateimarkierungen, Unterstützung für Reservierung, Variable Blockgröße und Datenkomprimierung vordefinierte Werte, die nicht geändert werden können.

Attribute für 1/4-Zoll-Bandlaufwerke mit 1200 MB (Typ 1200mb-c):

Im Folgenden sind die Attribute für 1/4-Zoll-Bandlaufwerke mit 1200 MB (Typ 1200mb-c) beschrieben.

Blockgröße

Die Standardblockgröße ist 512. Die anderen gültigen Blockgrößen sind 0 für Blöcke variabler Länge und 1024.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Eine Schreiboperation auf ein 1/4-Zoll-Band ist nur am Bandanfang (BOT, Beginning of Tape) oder nach dem Erkennen eines leeren Bandes möglich. Wenn das Band Daten enthält, können Sie diese Daten nur überschreiben, wenn sich der Schreibkopf am Bandanfang befindet. Wenn Sie einem Band Daten hinzufügen möchten, das beschrieben und anschließend zurückgespult wurde, müssen Sie bis zur nächsten Dateimarkierung vorspulen, woraufhin das System einen Fehler zurückgibt. Erst dann können Sie erneut mit dem Schreiben beginnen.

Neu spannen

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Die folgenden Einstellungen sind gültig:

| Einstellung | Bedeutung |
|-------------|---|
| 21 | QIC-1000* |
| 17 | QIC-525* |
| 16 | QIC-150 |
| 15 | QIC-120 |
| 0 | Standardeinstellung (QIC-1000) bzw. die letzte Einstellung für die Schreibdichte, die von einem System verwendet wurde. |

* QIC-525 und QIC-1000 sind die einzigen Modi, die die Blockgröße 1024 unterstützen.

Die Standardwerte sind 21 für Einstellung für Schreibdichte #1 und 17 für Einstellung für Schreibdichte #2.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 1/4-Zoll-Bandlaufwerk mit 1200 MB konfiguriert ist, haben die Attribute Erweiterte Dateimarkierungen, Unterstützung für Reservierung, Variable Blockgröße und Datenkomprimierung vordefinierte Werte, die nicht geändert werden können.

Attribute für 4-mm-Bandlaufwerke mit 12000 MB (selbst-konfigurierend):

Im Folgenden sind die Attribute für 4-mm-Bandlaufwerke mit 12000 MB (selbst-konfigurierend) beschrieben.

Blockgröße

Der Durchsatz des IBM 4-mm-Bandlaufwerks mit 12000 MB richtet sich maßgeblich nach der Blockgröße. Die empfohlene Mindestblockgröße für dieses Laufwerk sind 32 KB. Jede Blockgröße kleiner als 32 KB wirkt sich negativ auf die Übertragungsgeschwindigkeit (Sicherungs-/Wiederherstellungszeit) aus. In der folgenden Liste sind die empfohlenen Blockgrößen nach Befehl aufgelistet:

| Unterstützter Befehl | Standardblockgröße (Bytes) | Empfehlung |
|----------------------|--|---|
| BACKUP | 32 K oder 51,2 K (Standardeinstellung) | In Abhängigkeit davon, ob die Sicherung nach Namen durchgeführt wird, werden entweder 32 K oder 51,2 K verwendet. Es ist keine Änderung erforderlich. |
| TAR | 10 K | Die im Handbuch angegebene Blockgröße von 512 KB ist falsch. Setzen Sie den Blockungsparameter auf -N64 . |
| MKSYSB | Siehe BACKUP. | Der Befehl MKSYSB verwendet den Befehl BACKUP . Es ist keine Änderung erforderlich. |
| DD | | Setzen Sie den Blockungsparameter auf bs=32K . |
| CPIO | | Setzen Sie den Blockungsparameter auf -C64 . |

Anmerkung: Berücksichtigen Sie bei der Auswahl einer Blockgröße die Kapazität und den Durchsatz. Geringe Blockgrößen können einen erheblichen Einfluss auf die Leistung und einen minimalen Einfluss auf die Kapazität haben.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Die folgende Tabelle zeigt die unterstützten Einstellungen für Datenkassettyp und Schreibdichte (in Dezimal- und Hexadezimalschreibweise) für das IBM 4-mm-Bandlaufwerk mit 12000 MB. Wenn Sie eine Wiederherstellungsoperation (Lesen) durchführen, stellt das Bandlaufwerk die Schreibdichte automatisch auf die beim Schreiben gewählte Schreibdichte ein. Wenn Sie eine Sicherungsoperation (Schreiben) durchführen, müssen Sie die Einstellung für die Schreibdichte wählen, die für die verwendete Datenkassette richtig ist.

| Unterstützte Datenkassetten | Native Kapazität | Datenkapazität bei Komprimierung | SMIT-Einstellung für Schreibdichte | Einstellung für Schreibdichte in Hexadezimalschreibweise |
|-----------------------------|------------------|----------------------------------|------------------------------------|--|
| DDS III | 2,0 GB | 4,0 GB | 19 | 13h |
| DDS2 | 4,0 GB | 8,0 GB | 36 | 24h |
| DDS3 | 12,0 GB | 24,0 GB | 37 | 25h |

Anmerkung: Wenn Sie eine nicht unterstützte native Kapazität für die Datenkapazität anfordern, verwendet das Laufwerk standardmäßig die höchste unterstützte Kapazität für die Datenkassette, die in das Laufwerk eingelegt ist.

Datenkomprimierung

Die tatsächliche Komprimierung richtet sich nach dem Typ der geschriebenen Daten (siehe Tabelle oben). Für die Datenkapazität bei Komprimierung wird ein Komprimierungsverhältnis von 2:1 angenommen.

Attribute mit festen Werten

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Attribute für 1/4-Zoll-Bandlaufwerke mit 13000 MB (selbst-konfigurierend):

Im Folgenden sind die Attribute für 1/4-Zoll-Bandlaufwerke mit 13000 MB (selbst-konfigurierend) beschrieben.

Blockgröße

Die Standardblockgröße ist 512. Die anderen gültigen Blockgrößen sind 0 für Blöcke variabler Länge und 1024.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Eine Schreiboperation auf ein 1/4-Zoll-Band ist nur am Bandanfang (BOT, Beginning of Tape) oder nach dem Erkennen eines leeren Bandes möglich. Wenn das Band Daten enthält, können Sie diese Daten nur überschreiben, wenn sich der Schreibkopf am Bandanfang befindet. Wenn Sie einem Band Daten hinzufügen möchten, das beschrieben und anschließend zurückgespult wurde, müssen Sie bis zur nächsten Dateimarkierung vorspulen, woraufhin das System einen Fehler zurückgibt. Erst dann können Sie erneut mit dem Schreiben beginnen.

Neu spannen

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Die folgenden Einstellungen sind gültig:

| Einstellung | Bedeutung |
|--------------------|------------------------------------|
| 33 | QIC-5010-DC* |
| 34 | QIC-2GB* |
| 21 | QIC-1000* |
| 17 | QIC-525* |
| 16 | QIC-150 |
| 15 | QIC-120 |
| 0 | Standardeinstellung (QIC-5010-DC)* |

* QIC-525, QIC-1000, QIC-5010-DC und QIC-2GB sind die einzigen Modi, die die Blockgröße 1024 unterstützen.

Die Standardwerte sind 33 für Einstellung für Schreibdichte #1 und 34 für Einstellung für Schreibdichte #2.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 1/4-Zoll-Bandlaufwerk mit 13000 MB konfiguriert ist, haben die Attribute **Erweiterte Dateimarkierungen**, **Unterstützung für Reservierung** und **Variable Blockgröße** vordefinierte Werte, die nicht geändert werden können.

Attribute für 1/2-Zoll-Bandlaufwerke mit 9 Spuren (Typ 9trk):

Im Folgenden sind die Attribute für 1/2-Zoll-Bandlaufwerke mit 9 Spuren (Typ 9trk) beschrieben.

Blockgröße

Die Standardblockgröße ist 1024.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Die folgenden Einstellungen sind gültig:

| Einstellung | Bedeutung |
|-------------|-------------------------------------|
| 3 | 6250 Bit per Inch (bpi) |
| 2 | 1600 bpi |
| 0 | Die zuvor verwendete Schreibdichte. |

Die Standardwerte sind 3 für die Einstellung für Schreibdichte #1 und 2 für die Einstellung für Schreibdichte #2.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als 1/2-Zoll-Bandlaufwerk mit 9 Spuren konfiguriert ist, haben die Attribute Erweiterte Dateimarkierungen, Neu spannen, Unterstützung für Reservierung, Variable Blockgröße und Datenkomprimierung vordefinierte Werte, die nicht geändert werden können.

Attribute für 1/2-Zoll-Kassetten des Typs 3490e:

Im Folgenden sind die Attribute für die 1/2-Zoll-Kassette des Typs 3490e beschrieben.

Blockgröße

Die Standardblockgröße ist 1024. Dieses Laufwerk weist eine hohe Datenübertragungsrate auf, und die Blockgröße kann für einen effizienten Betrieb von entscheidender Bedeutung sein. Mit großen Blockgrößen kann die Betriebsgeschwindigkeit erheblich verbessert werden. Deshalb sollte generell die größtmögliche Blockgröße verwendet werden.

Anmerkung: Eine Erhöhung des Blockwertes kann zu Inkompatibilitäten mit anderen Programmen auf Ihrem System führen. In diesem Fall erhalten Sie beim Ausführen dieser Programme die folgende Fehlermeldung:

Ein Systemaufruf hat einen ungültigen Parameter empfangen.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Komprimierung

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Autoloader

Dieses Laufwerk besitzt einen Band-Sequencer, einen Autoloader, der nacheinander mehrere Bandkassetten aus dem Kassetten-Loader lädt und auswirft. Damit diese Funktion ordnungsgemäß ausgeführt wird, muss der Schalter an der Frontverkleidung auf die Position AUTO und das Attribut Autoloader auf ja eingestellt sein.

Attribute für andere SCSI-Bänder (Typ ost):

Im Folgenden sind Attribute für andere SCSI-Bänder (Typ ost) beschrieben.

Blockgröße

Der Systemstandardwert ist 512, aber dieser Wert muss an die Standardblockgröße für Ihr Bandlaufwerk angepasst werden. Typische Werte sind 512 und 1024. 8-mm- und 4-mm-Bandlaufwerke verwenden gewöhnlich 1024, und es wird Platz auf dem Band verschwendet, wenn das Attribut für die Blockgröße auf den Standardwert 512 eingestellt bleibt. Der Wert 0 gibt für einige Laufwerke eine variable Blockgröße an.

Einheitenpuffer

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Erweiterte Dateimarkierungen

Für diesen Typ von Bandlaufwerk gelten die allgemeinen Informationen für dieses Attribut.

Einstellung für Schreibdichte #1 und Einstellung für Schreibdichte #2

Der Standardwert ist für beide Einstellungen 0. Andere Werte und ihre Bedeutungen können für unterschiedliche Bandlaufwerke variieren.

Unterstützung für Reservierung

Der Standardwert ist nein. Sie können dieses Attribut auf ja setzen, wenn das Laufwerk Befehle für Reservierung und Freigabe unterstützt. Wenn Sie unsicher sind, sollten Sie den Wert nein verwenden.

Variable Blockgröße

Der Standardwert für die variable Blockgröße ist 0. Werte ungleich null werden hauptsächlich für Laufwerke mit 1/4-Zoll-Kassetten (QIC, Quarter Inch Cartridge) verwendet. Empfehlungen zur Einstellung dieses Attributs finden Sie in der SCSI-Spezifikation für das jeweilige Bandlaufwerk.

Wiederholungsintervall

Dieses Attribut gilt ausschließlich für Bandlaufwerke des Typs ost.

Zeitlimit für Lese-/Schreibvorgänge

Dieses Attribut gilt ausschließlich für Bandlaufwerke des Typs ost.

Attribute mit festen Werten

Wenn ein Bandlaufwerk als anderes SCSI-Bandlaufwerk konfiguriert ist, haben die Attribute erweiterte Dateimarkierungen, Neu spannen und Datenkomprimierung vordefinierte Werte, die nicht geändert werden können.

Attribute für MPIO-Bänder

MPIO-unterstützte Bänder haben zusätzliche Attribute, die unter "Attribute für MPIO-Einheiten" aufgelistet werden.

Zugehörige Konzepte:

„Multiple Path I/O“ auf Seite 575

Mit Multiple Path I/O (MPIO) kann eine Einheit über eine oder mehrere physische Verbindungen oder *Pfade* eindeutig erkannt werden.

Gerätedateien für Bandlaufwerke

Jedem Bandlaufwerk, das dem Betriebssystem bekannt ist, sind mehrere Gerätedateien zugeordnet.

Für das Schreiben und Lesen von Dateien auf Bandlaufwerken werden rmt-Gerätedateien verwendet. Diese Gerätedateien sind /dev/rmt*, /dev/rmt*.1, /dev/rmt*.2 bis /dev/rmt*.7. rmt* ist der logische Name eines Bandlaufwerks, z. B. rmt0, rmt1 usw.

Wenn Sie eine der Gerätedateien auswählen, die einem Bandlaufwerk zugeordnet sind, treffen Sie damit Festlegungen in Bezug darauf, wie die E/A-Operationen, an denen das Bandlaufwerk beteiligt ist, ausgeführt werden.

| Eintrag | Beschreibung |
|-------------------|---|
| Dichte | Sie können auswählen, ob mit der Einstellung für Schreibdichte #1 oder #2 auf das Bandlaufwerk geschrieben wird. Die Werte für diese Dichteeinstellungen gehören zu den Attributen des Bandlaufwerks. Da die Einstellung für Schreibdichte #1 gewöhnlich auf die höchstmögliche Dichte für das Bandlaufwerk und die Einstellung für Schreibdichte #2 auf die nächstmögliche Dichte gesetzt wird, werden Gerätedateien, die die Einstellung für Schreibdichte #1 verwenden, gelegentlich auch mit HD (High Density, hohe Dichte) bezeichnet, und die Gerätedateien, die die Einstellung für Schreibdichte #2 verwenden, mit LD (Low Density, geringe Dichte) aber diese Sicht ist nicht immer korrekt. Beim Lesen von einem Band wird die Dichteeinstellung ignoriert. |
| Rewind-on-Close | Sie können auswählen, ob das Band zurückgespult wird, wenn die Gerätedatei, die auf das Bandlaufwerk verweist, geschlossen wird. Wenn Sie die Option "rewind-on-close" auswählen, wird das Band am Bandanfang positioniert, wenn die Datei geschlossen wird. |
| Retension-on-Open | Sie können auswählen, ob das Band neu gespannt wird, wenn die Datei geöffnet wird. "Neu spannen" bedeutet, dass das Band an das Ende vorgespult und anschließend wieder an den Anfang zurückgespult wird, um Fehler zu reduzieren. Wenn Sie die Option "retension-on-open" auswählen, wird das Band am Bandanfang positioniert, wenn die Datei geöffnet wird. |

Die folgende Tabelle enthält die Namen der rmt-Gerätedateien und ihre Merkmale.

| Gerätedatei | Rewind-on-Close | Retension-on-Open | Einstellung für Schreibdichte |
|-------------|-----------------|-------------------|-------------------------------|
| /dev/rmt* | Ja | Nein | #1 |
| /dev/rmt*.1 | Nein | Nein | #1 |
| /dev/rmt*.2 | Ja | Ja | #1 |
| /dev/rmt*.3 | Nein | Ja | #1 |
| /dev/rmt*.4 | Ja | Nein | #2 |
| /dev/rmt*.5 | Nein | Nein | #2 |
| /dev/rmt*.6 | Ja | Ja | #2 |
| /dev/rmt*.7 | Nein | Ja | #2 |

Angenommen, Sie möchten drei Dateien auf das Band im Bandlaufwerk rmt2 schreiben. Die erste Datei soll am Bandanfang, die zweite Datei nach der ersten Datei und die dritte Datei nach der zweiten Datei geschrieben werden. Nehmen wir weiter an, dass Sie die Einstellung für Schreibdichte #1 für das Bandlaufwerk verwenden möchten. Die in der folgenden Liste aufgeführten Gerätedateien könnten in der angegebenen Reihenfolge für das Beschreiben des Bands verwendet werden:

1. /dev/rmt2.3
2. /dev/rmt2.1
3. /dev/rmt2

Die Gründe für die Auswahl dieser speziellen Gerätedateien sind im Folgenden erläutert:

- /dev/rmt2.3 wird als erste Datei ausgewählt, weil diese Datei das Merkmal "Retension-on-Open" hat, das sicherstellt, dass die erste Datei am Anfang des Bands geschrieben wird. Rewind-on-Close wird nicht ausgewählt, weil die nächste E/A-Operation dort beginnen soll, wo diese Datei endet. Wenn sich das Band beim Öffnen der ersten Datei bereits am Bandanfang befindet, wäre die Datei /dev/rmt2.1 die schnellere Wahl, weil die Zeit für das Neuspannen des Bandes entfällt.
- /dev/rmt2.1 wird als zweite Datei ausgewählt, weil bei dieser Datei weder Retension-on-Open noch Rewind-on-Close ausgewählt ist. Es gibt keinen Grund, an den Bandanfang zurückzuspulen, wenn die Datei geöffnet oder geschlossen wird.
- /dev/rmt2 wird als dritte und letzte Datei ausgewählt, weil Retension-on-Open nicht erwünscht ist, da die dritte Datei im Anschluss an die zweite Datei geschrieben werden soll. Rewind-on-Close ist ausgewählt, weil nicht geplant ist, nach der dritten Datei weitere Daten auf das Band zu schreiben. Bei der nächsten Verwendung des Bandes wird das Band neu beschrieben.

Neben der Steuerung von Bandoperationen durch Auswahl einer bestimmten rmt-Gerätedatei können Sie den Befehl `tctl` zur Steuerung von Bandoperationen verwenden.

| Unterstützung von USB-Einheiten

| Das Betriebssystem AIX unterstützt USB-Einheiten (Universal Serial Bus).

| Das Betriebssystem AIX unterstützt die folgenden Typen von USB-Einheiten anderer Anbieter und IBM USB-Einheiten:

- | • Flashlaufwerk
- | • Plattenlaufwerk
- | • Optisches Laufwerk (Blu-ray, DVD und CD-ROM)
- | • Bandlaufwerk
- | • Tastatur
- | • Maus
- | • Lautsprecher

| **Anmerkung:** Das Betriebssystem AIX unterstützt keine USB-Drucker.

| Einige USB-Einheiten anderer Anbieter werden vom Betriebssystem AIX möglicherweise nicht erkannt. Es könnte möglicherweise nicht genügend elektrischer am Power Systems-USB-Port verfügbar sein. Deshalb unterstützt das Betriebssystem AIX nicht alle verfügbaren USB-Einheiten anderer Anbieter.

| **Unterstützung von USB-Flashlaufwerken**

| Ab AIX 5.3 mit Technology Level 5300-09 und AIX 6.1 mit Technology Level 6100-02 werden USB-Flashlaufwerke (Universal Serial Bus) unterstützt.

| Die Unterstützung für diese Einheiten sind im folgenden Einheitenpaket enthalten:

| `devices.usbif.08025002`

| Die AIX-Unterstützung für USB-Flashlaufwerke wurde mit einer Stichprobe von OEM-USB-Flashlaufwerken geprüft, die dem Branchenstandard entsprechen. Die Einheitentreiber für den AIX-USB-Host-Controller unterstützen USB 2.0. USB-Flashlaufwerke werden mit logischen Namen wie `usbms0` und `usbms1` konfiguriert und stellen Rohgerätedateien und blockorientierte Gerätedateien dar. Die Rohgerätedatei für `usbms0` ist beispielsweise `/dev/rusbms0`, und die blockorientierte Gerätedatei ist `/dev/usbms0`. Vor AIX Version 5.3 mit Technology Level 5300-11 und AIX Version 6.1 mit Technology Level 6100-04 wurden USB-Flashlaufwerke als `/dev/flashdrive0` konfiguriert.

| Auf diesen Laufwerken wird das International Organization for Standardization-Dateisystem (ISO 9660, schreibgeschützt) unterstützt. Sie können eine Systemsicherung auf Laufwerken mit dem Befehl `tar`, mit dem Befehl `cpio` oder mit den Sicherungs- und Wiederherstellungsarchiven erstellen. Außerdem können Sie die ISO-Images mit dem Befehl `dd` den Laufwerken hinzufügen.

| Das Betriebssystem AIX unterstützt die Plug-and-play-Funktionalität für USB-Flashlaufwerke nicht. Um AIX-Benutzern ein Flashlaufwerk zur Verfügung zu stellen, muss ein Rootbenutzer das Laufwerk mit einem USB-Anschluss des Systems verbinden und den folgenden Befehl ausführen:

| `cfgmgr -l usb0`

| **Achtung:** Gehen Sie beim Entfernen der Flashlaufwerke von Anschlüssen vorsichtig vor. Wenn die Laufwerke nicht ordnungsgemäß geschlossen oder abgehängt werden, bevor Sie sie entfernen, können die Daten auf den Laufwerken beschädigt werden.

| Nach dem Entfernen der Laufwerke bleiben diese so lange in der ODM-Datenbank (Object Data Manager) verfügbar, bis der Rootbenutzer den folgenden Befehl ausführt:

| `rmdev -l usbmsn`

| Wenn ein Laufwerk verfügbar ist, können Sie es erneut mit dem System verbinden und dann erneut anhängen oder öffnen. Wenn ein Laufwerk von einem USB-Anschluss des Systems getrennt wird, während es für einen Benutzer geöffnet ist, ist dieses Laufwerk erst wieder verwendbar, wenn der Benutzer das Laufwerk schließt und dann erneut öffnet.

| **Leseunterstützung für USB-Blu-ray-Laufwerke**

| AIX Version 6.1 mit 6100-06 Technology Level und höher erkennt und konfiguriert über USB angeschlossene Blu-ray-Laufwerke.

| Dieses Feature ist im folgenden Einheitenpaket enthalten:

| `devices.usbif.08025002`

| Die Fähigkeit des Betriebssystems AIX, Blu-ray-Medien zu lesen, wurde mithilfe einer Stichprobe von OEM-USB-Blu-ray-Laufwerken geprüft, die dem Branchenstandard entsprechen.

| USB-Blu-ray-Laufwerke werden mithilfe logischer Namen wie `cd0` und `cd1` konfiguriert. Die Laufwerke stellen Rohgerätedateien und blockorientierte Gerätedateien dar. Die Rohgerätedatei für `cd0` ist beispielsweise `/dev/rcd0`, und blockorientierte Gerätedatei ist `/dev/cd0`.

| Lesefunktionen werden für das International Organization for Standardization-Dateisystem (ISO 9660, schreibgeschützt), das Universal Disk Format-Dateisystem (UDF Version 2.01 und früher) und für Standardbefehle für den Zugriff auf optische Medien wie **dd** und **tar** unterstützt.

| Das Betriebssystem AIX unterstützt keine Schreiboperationen auf CD-, DVD- und Blu-ray-Medien im USB-Blu-ray-Laufwerk. Obwohl die Schreiboperation nicht verhindert wird (falls das Schreiben auf das Laufwerk möglich ist), bietet IBM keine Unterstützung für Probleme, die während der Schreiboperation auftreten.

| Das Betriebssystem AIX unterstützt die Plug-and-play-Funktionalität für USB-Blu-ray-Laufwerke nicht. Um AIX-Benutzern ein USB-Blu-ray-Laufwerk zur Verfügung zu stellen, muss ein Rootbenutzer das Laufwerk mit dem USB-Anschluss des Systems verbinden und den folgenden Befehl ausführen:

```
| cfgmgr -l usb0
```

| Nach dem Entfernen des Laufwerks bleibt das Laufwerk so lange in der ODM-Datenbank (Object Data Manager) verfügbar, bis der Rootbenutzer den folgenden Befehl ausführt:

```
| rmdev -l cdn
```

| Wenn ein Laufwerk verfügbar ist, können Sie es erneut mit dem System verbinden. Wenn ein Laufwerk von einem System-USB-Anschluss getrennt wird, während es für einen Benutzer noch geöffnet ist, können Sie dieses Laufwerk erst verwenden, wenn Sie es schließen und dann erneut öffnen.

Caching von Speicherdaten

Das serverseitige Caching von Speicherdaten wird auf den Cacheeinheiten unterstützt.

Als Cacheeinheiten können die folgenden Typen von Einheiten verwendet werden:

- An den Server angeschlossene Flascheinheiten wie integrierte Solid-State-Laufwerke im Server
- Flascheinheiten, die über SAS-Controller (Serial Attached SCSI) direkt an den Server angeschlossen sind
- Flashressourcen im Speicherbereichsnetz (SAN)

Konzept für das Caching von Speicherdaten

Sie können die Speicherdaten dynamisch zwischenspeichern (Caching starten und stoppen), während die Workload verarbeitet wird. Die Workload muss nicht inaktiviert werden, um die Caching-Operation auszuführen.

Für die Beschreibung des Caching-Konzepts werden die folgenden Begriffe verwendet:

Cache-Einheit

Eine Cache-Einheit ist ein Solid-State-Laufwerk (SSD, Solid-State Drive) oder eine Flashplatte, die für das Caching verwendet wird.

Cache-Pool

Ein Cache-Pool ist eine Gruppe von Cache-Einheiten, die ausschließlich für das Speichercaching verwendet werden.

Cachepartition

Eine Cachepartition ist eine logische Cache-Einheit, die aus dem Cache-Pool erstellt wird.

Zieleinheit

Eine Zieleinheit ist eine Speichereinheit, die zwischengespeichert wird.

Zum Zwischenspeichern einer oder mehrerer Zieleinheiten kann eine einzige Cachepartition verwendet werden. Wenn eine Zieleinheit zwischengespeichert wird, werden alle Leseanforderungen für die Einheitenblöcke an die Caching-Software weitergeleitet. Wird ein bestimmter Block im Cache gefunden, wird

die E/A-Anforderung über die Cache-Einheit verarbeitet. Wenn der angeforderte Block nicht im Cache gefunden wird oder wenn es sich um eine Schreibanforderung handelt, wird die E/A-Anforderung an die Zieleinheit zurückgegeben.

Vorteile des Cachings von Speicherdaten

Durch das serverseitige Caching von Speicherdaten kann die Virtualisierungsdichte erhöht werden, insbesondere bei einer Überlastung des Speichersubsystems.

Das Caching von Speicherdaten hat die folgenden Vorteile:

Latenzzeit

Analyse- und transaktionsorientierte Workloads haben geringere Abfrageantwortzeiten, weil der SSD-Speicher (Solid-State drive, Solid-State-Laufwerk) geringere Latenzzeiten hat. Durch die Verwendung serverseitigen Cachings kann die durchschnittliche Latenzzeit für eine transaktionsorientierte Workload um die Hälfte reduziert werden.

Durchsatz

OLTP-Workloads (Online Transaction Processing, Onlinetransaktionsverarbeitung) haben höhere Transaktionsraten, weil der SSD-Speicher einen höheren Durchsatz hat.

Schreibdurchsatz

In Umgebungen, in denen das Speicherbereichsnetz (SAN, Storage Area Network) überlastet ist, kann die als Cache verwendete Flascheinheit einen erheblichen Prozentsatz der Leseanforderungen auslagern. Durch die Auslagerung der Leseanforderungen kann sich der Schreibdurchsatz des SAN verbessern und das SAN kann effektiv mehr Clients und Hosts bedienen.

Geringerer Speicherbedarf

Wenn eine Flash-Cache-Einheit konfiguriert ist, können einige Workloads sogar mit einem geringeren Speicherbedarf verarbeitet werden.

Einschränkungen beim Caching von Speicherdaten

Stellen Sie sicher, dass Sie mit den Einschränkungen und den zusätzlichen Konfigurationsanforderungen für die Verwendung des Caching-Features vertraut sind. Außerdem müssen Sie die Anwendungseinschränkungen für die zwischenzuspeichernden Zieleinheiten berücksichtigen.

Berücksichtigen Sie die folgenden Einschränkungen für das Caching der Speicherdaten:

- Die Caching-Software ist als schreibgeschützter Cache konfiguriert, d. h., es werden nur Leseanforderungen vom Flash-SSD (Solid-State Drive, Solid-State-Laufwerk) verarbeitet. Alle Schreibanforderungen werden von der ursprünglichen Speichereinheit verarbeitet.
- Auf die Speichereinheit geschriebene Daten werden nicht automatisch in den Cache gefüllt. Wenn die Schreiboperation für einen im Cache befindlichen Block ausgeführt wird, werden die vorhandenen Daten im Cache als ungültig markiert. Je nach Häufigkeit und Aktualität des Zugriffs auf den Block erscheint derselbe Block im Cache erneut im Cache.
- In jeder AIX-LPAR (Logical Partition, logische Partition) ist zusätzlicher Speicher erforderlich, weil die Caching-Software Metadaten in jedem Leseblock verwaltet. Für jede LPAR, in der das Caching aktiviert ist, sind mindestens 4 GB Speicher erforderlich.
- Die Caching-Software lädt Daten basierend auf lokalen Lesemustern in den Cache und macht die Cacheeinträge lokal ungültig. Die Zieleinheiten dürfen jeweils nur von einer einzigen LPAR genutzt werden. Die Zieleinheiten dürfen nicht zu einem Clusterspeicher wie Oracle Real Application Clusters (RAC), DB2 pureScale oder General Parallel File System (GPFS) gehören. Zieleinheiten, die zu einem Hochverfügbarkeitscluster gehören, können nur zwischengespeichert werden, wenn der Zugriff festlegt, dass jeweils nur ein einziger Host Daten auf die Zieleinheit schreiben bzw. von der Zieleinheit lesen darf, und das Caching nur auf dem aktiven Knoten aktiviert ist.
- Die Cacheplatte kann einer AIX-LPAR oder einer VIOS-LPAR (Virtual I/O Server) bereitgestellt werden. Cacheeinheiten dürfen nicht gemeinsam genutzt werden.
- Die Caching-Software muss die Zieleinheiten öffnen, um alle E/A-Anforderungen an die Zieleinheiten abzufangen. Wenn eine Workload nach dem Start des Cachings die Zieleinheit exklusiv öffnen muss,

schlägt diese Operation zum exklusiven Öffnen fehl. In solchen Fällen muss das Caching gestoppt und nach dem Start der Workload erneut gestartet werden.

- Wenn die Platten als Zieleinheiten verwendet werden, darf das Attribut **reserve_policy** der Platten nicht auf `single_path` gesetzt sein.
- Wenn die Caching-Operation für eine Zieleinheit gestartet wird, verzögert die Logik der Cache-Engine die Umstufung der Daten in den Cache. Diese Verzögerung ist erforderlich, um sicherzustellen, dass alle ausstehenden E/A-Operationen für die Zieleinheit, die vor dem Start der Caching-Operation abgesetzt werden, abgeschlossen sind, bevor die Caching-Operation gestartet wird. Die exakte Verzögerungszeit wird intern auf der Basis der Anzahl verfügbarer Pfad und des Attributs **rw_timeout** (sofern vorhanden) der Zielplatte berechnet. Wenn die intern berechnete Zeit durch eine benutzerdefinierte Zeit überschrieben werden muss, können Sie die Umgebungsvariable `DEFAULT_IO_DRAIN_TIMEOUT_PD` in der Datei `/etc/environment` auf einen angepassten Zeitlimitwert (in Sekunden) setzen.

Komponenten des Speicherdatencachings

Die Caching-Software setzt sich aus den Komponenten "Cache-Management" und "Cache-Engine" zusammen.

Cache-Management

Sie können das Speicherdatencaching mit dem Befehl **cache_mgt** verwalten, der im Betriebssystem AIX und in Virtual I/O Server (VIOS) verfügbar ist. Mit dem Befehl **cache_mgt** können Sie die folgenden Tasks ausführen:

- Cache-Pool erstellen und partitionieren
- Cachpartition einer Zieleinheit oder der logischen AIX-Partition (LPAR) als vSCSI-Einheit (Virtual Small Computer System Interface) zuweisen
- Caching-Operation starten und stoppen

Cache-Engine

Die Cache-Engine ist die wichtigste Komponente der Caching-Software. Die Cache-Engine entscheidet, welche Blöcke im Speicher zwischengespeichert werden müssen und ob die Daten aus dem Cache oder aus dem primären Speicher abgerufen werden müssen.

Der Caching-Algorithmus basiert auf einem so genannten Populate-on-Read-Mechanismus, der den Cache mit Daten mit räumlicher Lokalität füllt (Daten, die sich in der Nähe von Blöcken befinden, die erst kürzlich gelesen wurden). Der Caching-Algorithmus füllt Daten schneller in den Cache, wenn der Cache leer ist.

Alle Blöcke im Cache werden überwacht, um zu prüfen, wie oft sie gelesen werden, und es wird eine so genannte Heat-Map (Wärmebild) generiert. In der Heat-Map werden sowohl Zugriffshäufigkeit als auch Zugriffsaktualität berücksichtigt. Wenn der Cache vollständig gefüllt ist, werden dem Cache nur dann neue Einträge hinzugefügt, wenn der neue Block "wärmer" ist als der "kälteste" Block im Cache. Der kälteste Block wird aus dem Cache entfernt und der neue Eintrag wird hinzugefügt.

Das aggressive Befüllen des Cache gewährleistet kurze Warm-up-Zeiten, die den Cache effektiv machen, sobald er aktiviert ist. Die Richtlinie für das Entfernen von Daten aus dem Cache, die auf der Heat-Map basiert, stellt sicher, dass das Caching dynamisch erfolgt und sich an sich ändernde Workloadmuster anpasst.

Caching von Speicherdaten konfigurieren

Im Betriebssystem AIX wird das serverseitige Caching von Flascheinheiten in mehreren individuellen Konfigurationen unterstützt. Diese Konfigurationen unterscheiden sich durch die Art der Bereitstellung der Cache-Einheit für die AIX-LPAR (Logical Partition, logische Partition).

Das serverseitige Caching unterstützt die folgenden Modi im Betriebssystem AIX:

- Dedizierter Modus
- Virtueller Modus
- NPIV-Modus (N_Port ID Virtualization)

Caching von Speicherdaten im dedizierten Modus:

Im dedizierten Modus wird die Cache-Einheit der AIX-LPAR direkt bereitgestellt.

Sie müssen einen Cache-Pool und anschließend eine Cache-Partition auf der Cache-Einheit erstellen. Auf einer dedizierten Cache-Einheit kann nur eine einzige Cachepartition erstellt werden. Sie können die Cachepartition verwenden, um eine beliebige Anzahl von Zieleinheiten in der AIX-LPAR zwischenspeichern. Die LPAR ist nicht mobil, weil die Cacheeinheit dieser LPAR zugeordnet ist. Wenn die LPAR auf einen anderen Server migriert werden muss, müssen Sie das Caching manuell stoppen und die Cache-Einheit vor der Migration dekonfigurieren.

Die folgende Abbildung zeigt ein Beispiel für eine Caching-Konfiguration in einer AIX-LPAR für eine dedizierte Cache-Einheit.

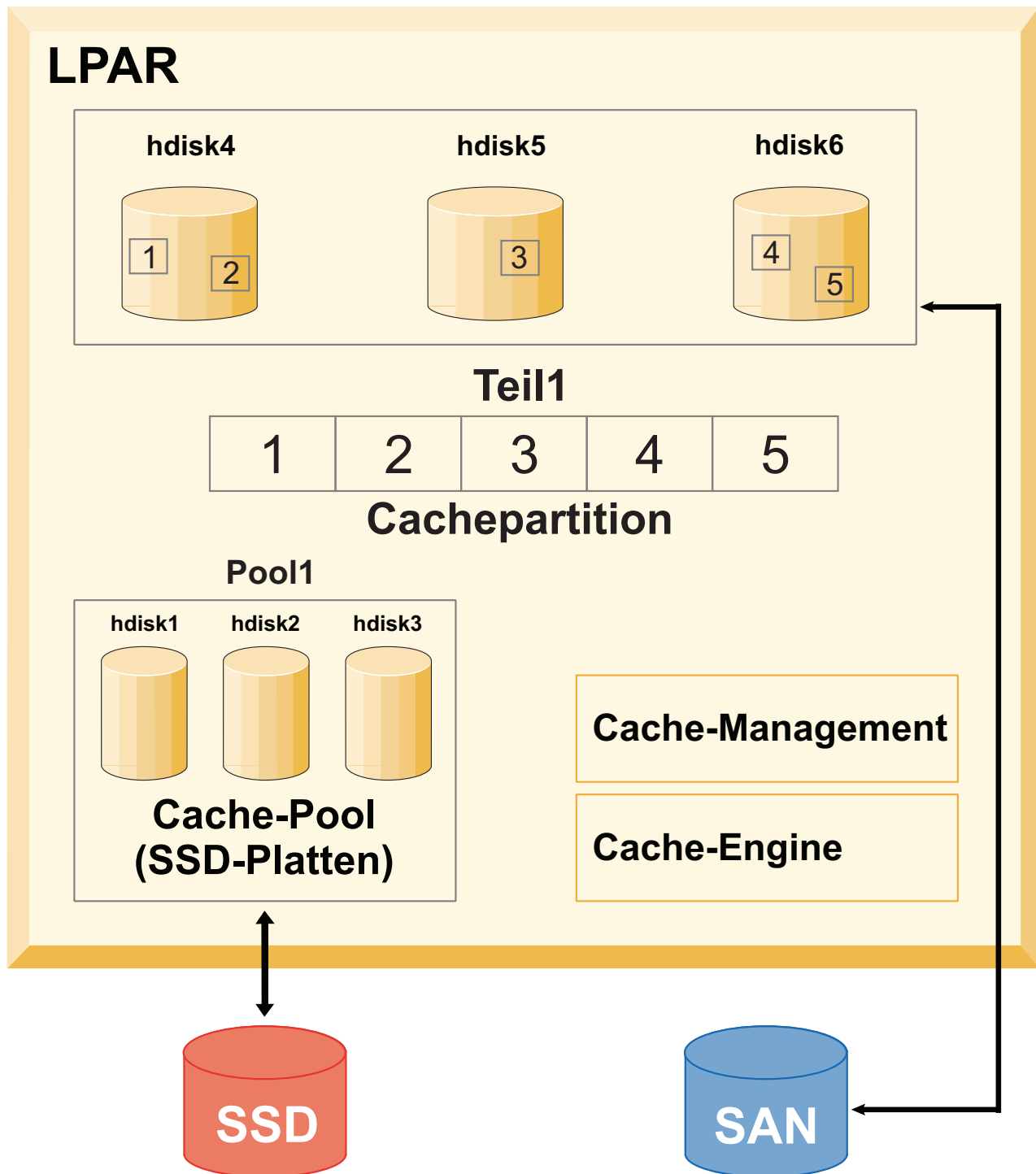


Abbildung 15. Caching von Speicherdaten: Konfiguration für eine dedizierte Cache-Einheit

Angenommen, es gibt die Cache-Einheiten **hdisk1**, **hdisk2** und **hdisk3** und die Zieleinheiten **hdisk4**, **hdisk5** und **hdisk6**. Führen Sie die folgenden Schritte aus, um das Caching der Zieleinheiten zu starten und zu überwachen:

1. Erstellen Sie einen Cache-Pool im SSD-Speicher.

```
# cache_mgt pool create -d hdisk1,hdisk2,hdisk3 -p cmpool0
```
2. Erstellen Sie eine Cachepartition mit einer Größe von 80 MB aus dem Cache-Pool.

```
# cache_mgt partition create -p cmpool0 -s 80M -P part1
```

3. Weisen Sie die Cachepartition den Zielplatten zu, die Sie zwischenspeichern möchten.

```
# cache_mgt partition assign -t hdisk4 -P part1
# cache_mgt partition assign -t hdisk5 -P part1
# cache_mgt partition assign -t hdisk6 -P part1
```

4. Starten Sie das Caching der Zieleinheiten.

```
# cache_mgt cache start -t hdisk4
# cache_mgt cache start -t hdisk5
# cache_mgt cache start -t hdisk6
```

5. Überwachen Sie die Statistiken zu den Cachetreffern.

```
# cache_mgt monitor get -h -s
```

Zugehörige Informationen:

Befehl `cache_mgt`

Caching von Speicherdaten im virtuellen Modus:

Im virtuellen Modus wird die Cache-Einheit dem Virtual I/O Server (VIOS) zugeordnet.

Im virtuellen Modus wird der Cache-Pool im VIOS erstellt. Im VIOS wird der Cache-Pool dann in Partitionen eingeteilt. Jede Cachepartition kann einem vhost-Adapter (virtueller Host) zugewiesen werden.

Wenn die Cachepartition in der logischen AIX-Partition (LPAR) erkannt wird, kann sie für das Caching der Zieleinheit verwendet werden. Da es sich bei der Cache-Einheit um eine virtuelle Einheit handelt, kann die Cachepartition auf einen anderen Server migriert werden. Vor der Migration wird das Caching in der Quellen-LPAR automatisch gestoppt. Im Rahmen der Migrationsoperation wird im Ziel-VIOS dynamisch eine Cachepartition mit derselben Größe erstellt, wenn die Caching-Software installiert und ein Cache-Pool im Ziel-VIOS verfügbar ist. Während der Migration wird die Cachepartition für die LPAR verfügbar gemacht. Nach Abschluss der Migration wird automatisch das Caching in der Ziel-LPAR gestartet. In diesem Fall beginnt das Caching mit einem leeren (nicht mit Daten gefüllten) Cache.

In der folgenden Abbildung sehen Sie ein Beispiel für eine Caching-Konfiguration im virtuellen Modus. Die Cache-Einheit befindet sich in einer VIOS-LPAR und die Zieleinheit in einer AIX-LPAR.

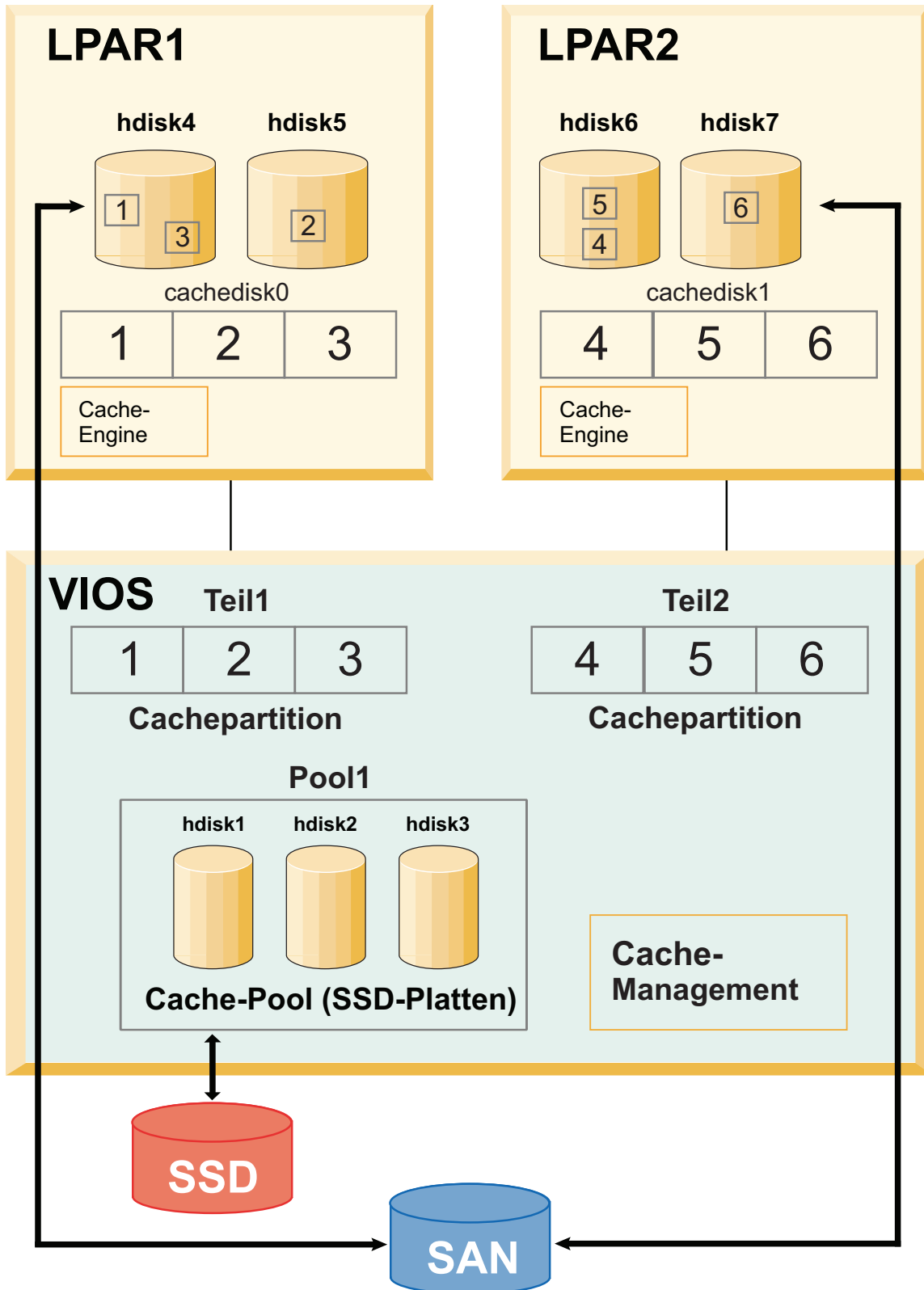


Abbildung 16. Caching von Speicherdaten: Konfiguration für eine virtuelle Cache-Einheit

Angenommen, die Cache-Einheiten sind **hdisk1**, **hdisk2** und **hdisk3** (in der VIOS-LPAR) und die Zieleinheiten sind **hdisk4** und **hdisk5** (in der AIX-LPAR). Gehen Sie wie folgt vor, um das Caching der Zieleinheiten zu starten und zu überwachen:

1. Erstellen Sie in der VIOS-LPAR unter Verwendung des SSD-Speichers einen Cache-Pool.
cache_mgt pool create -d hdisk1,hdisk2,hdisk3 -p cmpool0
2. Erstellen Sie in der VIOS-LPAR eine Cachepartition mit einer Größe von 80 MB aus dem Cache-Pool.
cache_mgt partition create -p cmpool0 -s 80M -P part1
3. Weisen Sie in der VIOS-LPAR die Partition einem vhost-Adapter zu.
cache_mgt partition assign -P part1 -v vhost0
4. Weisen Sie in der AIX-LPAR die Cachepartition Zieleinheiten zu.
cache_mgt partition assign -t hdisk4 -P cachedisk0
cache_mgt partition assign -t hdisk5 -P cachedisk0
5. Starten Sie in der AIX-LPAR das Caching der Zieleinheiten.
cache_mgt cache start -t hdisk4
cache_mgt cache start -t hdisk5
6. Überwachen Sie in der AIX-LPAR die Statistiken zu den Cachetreffern.
cache_mgt monitor get -h -s

Zugehörige Informationen:

Befehl cache_mgt

Befehl cache_mgt in VIOS

Caching von Speicherdaten im NPIV-Modus:

In diesem Modus ist die Cacheeinheit als virtuelle Fibre Channel-Einheit (N_Port ID Virtualization) in der logischen AIX-Partition (LPAR) verfügbar.

Sie müssen einen Cache-Pool und anschließend eine Cache-Partition in der AIX-LPAR erstellen. In der AIX-LPAR kann nur eine einzige Cachepartition erstellt werden. Sie können die Cachepartition verwenden, um eine beliebige Anzahl von Zieleinheiten in der AIX-LPAR zwischenspeichern. Die LPAR kann auf einen anderen Server migriert werden, weil die Cacheeinheit im Speicherbereichsnetz (SAN, Storage Area Network) verfügbar ist. Die Cacheeinheit muss auf dem Zielsystem sichtbar gemacht werden. Die Caching-Operation kann während des Migrationsprozesses fortgesetzt werden und der Cache wird nach Abschluss der Migration gefüllt.

Das Caching der Zieleinheit im NPIV-Modus ist dasselbe wie das Caching der Speicherdaten im dedizierten Modus mit der Ausnahme, dass die Cacheeinheit im SAN verfügbar ist. Die Prozedur für das Caching der Zieleinheit ist jedoch dieselbe wie die Prozedur für das Caching der Speicherdaten im dedizierten Modus.

Speicherdatencache verwalten

Obwohl der Cache konfiguriert ist, können sich die Caching-Anforderungen mit der Zeit ändern. Sie könnten beispielsweise neue zwischenspeichernde Workloads hinzufügen. Zur Erfüllung der Caching-Anforderungen können Sie den Cache-Pool mit zusätzlichen Cacheeinheiten erweitern, eine neue Cachepartition in einem vorhandenen Pool erstellen oder eine vorhandene Partition vergrößern.

Sie können zur Verwaltung der Caching-Konfiguration die folgenden Beispiele verwenden:

- Geben Sie den folgenden Befehl ein, um dem Pool eine Cacheeinheit hinzuzufügen:

```
# cache_mgt pool extend -p pool1 -d hdisk4 -f
```

Wenn Sie das Flag **-f** angeben, wird die vorhandene Nutzung der Platte (hdisk4) außer Kraft gesetzt, falls die Platte eine vorhandene Datenträgergruppe enthält.

- Geben Sie den folgenden Befehl ein, um eine Partition für eine neue Workload mit einer Größe von 100 MB zu erstellen:

```
# cache_mgt partition create -p pool1 -s 100M -P part2
```

- Geben Sie den folgenden Befehl ein, um die Größe einer vorhandenen Partition um 20 MB zu erhöhen:

```
# cache_mgt partition extend -p part1 -s 20M
```

Hinweise zur Hochverfügbarkeit:

Wenn die Zieleinheiten, die zwischengespeichert werden, zu einer Ressourcengruppe gehören, die in einem Hochverfügbarkeitscluster verwaltet wird, muss die Failover-Operation sorgfältig geplant werden.

Das Caching kann jeweils nur für einen einzigen Knoten aktiviert werden. Bevor ein Failover-Ereignis eingeleitet wird, müssen Sie sicherstellen, dass die Caching-Operation auf dem ursprünglichen System inaktiviert ist. Nach Abschluss des Failovers auf ein alternatives System müssen Sie die Caching-Software manuell aktiviert werden.

Gehen Sie zum Aktivieren der Caching-Software wie folgt vor:

1. Stoppen Sie das Caching auf dem ursprünglichen System.

```
# cache_mgt cache stop -t hdisk2
```

2. Starten Sie nach Abschluss der Fehlerbehebung das Caching auf dem neuen System.

```
# cache_mgt cache start -t hdisk2
```

Cachestatistiken überwachen

Die Cachestatistiken für jede Zieleinheit können mit dem Befehl **cache_mgt monitor get** angezeigt werden.

Wenn beispielsweise hdisk1 die einzige Zieleinheit ist, die zwischengespeichert wird, kann der Befehl **cache_mgt** eine Ausgabe wie die folgende haben:

```
# cache_mgt monitor get -h -s
ETS Device I/O Statistics -- hdisk1
Start time of Statistics -- Mon Mar 27 07:10:41 2017
-----
Read Count:                152125803
Write Count:                79353626
Read Hit Count:            871
Partial Read Hit Count:    63
Read Bytes Xfer:          10963365477376
Write Bytes Xfer:          4506245999616
Read Hit Bytes Xfer:       48398336
Partial Read Hit Bytes Xfer: 5768192
Promote Read Count:        2033078104
Promote Read Bytes Xfer:   532959226494976
```

Der Befehl **cache_mgt** stellt die folgenden Caching-Metriken bereit:

Read Count

Die Gesamtanzahl aller Leseoperationen, die von allen Anwendungen an die Zieleinheit abgesetzt werden. Dieser Wert beinhaltet auch die Leseoperationen an die Cacheeinheit, wenn die Daten in der Cacheeinheit verfügbar sind. Der Wert ist die Gesamtanzahl separater Leseanforderungen und gibt keinen Hinweis auf die Größe (Byteanzahl) der Leseanforderungen.

Write Count

Die Gesamtanzahl der Schreiboperationen, die an die Zieleinheit abgesetzt werden. Der Wert ist die Gesamtanzahl separater Schreibenanforderungen und gibt keinen Hinweis auf die Größe (Byteanzahl) der Schreibenanforderungen.

Read Hit Count

Die Gesamtanzahl der Leseoperationen, die an die Zieleinheit abgesetzt werden und *Volltreffer* sind. Ein *Lesevolltreffer* ist eine Leseoperationsinstanz, die vollständig von der Cacheeinheit erfüllt wird. Der Wert von **Read Hit Count** gibt die Gesamtanzahl separater Leseanforderungen mit Treffer an und gibt keinen Hinweis auf die Größe (Byteanzahl) der Leseanforderungen. Dieser Wert ist im Wert von **Read Count** enthalten.

Partial Read Hit Count

Die Gesamtanzahl der Leseoperationen, die an die Zieleinheit abgesetzt werden und *partielle Treffer* sind. Ein *partieller Treffer* ist eine Leseoperationsinstanz, die teilweise von der Cacheeinheit erfüllt wird. Die verbleibenden Daten, die nicht in der Cacheeinheit verfügbar sind, müssen von der Zieleinheit abgerufen werden. Der Wert von **Partial Read Hit Count** ist die Gesamtanzahl separater Leseanforderungen und gibt keinen Hinweis auf die Größe (Byteanzahl) der Leseanforderungen. Dieser Wert ist im Wert von **Read Count** enthalten.

Read Bytes Xfer

Die Gesamtanzahl der Bytes, die in allen Leseanforderungen übertragen werden, die von Anwendungen an die Zieleinheit abgesetzt wurden. Dieser Wert stellt die Gesamtbyteanzahl der Daten von *Volltreffern*, der Daten von *partiellen Treffern* und der Daten, die von der Zieleinheit abgerufen werden, dar.

Write Bytes Xfer

Die Gesamtanzahl der Bytes, die in allen Schreibanforderungen übertragen werden, die von Anwendungen an die Zieleinheit abgesetzt wurden.

Read Hit Bytes Xfer

Die Gesamtanzahl der Bytes, die bei *Volltreffern* gelesen werden.

Partial Read Hit Bytes Xfer

Die Gesamtanzahl der Bytes, die bei *partiellen Treffern* gelesen werden.

Promote Read Count

Die Gesamtanzahl der Lesebefehle, die an die Zieleinheit abgesetzt werden, wenn die Daten in die Cacheeinheit geschrieben werden. Dieser Wert gibt nicht die Anzahl der Instanzen an, bei denen die Daten in die Cacheeinheit geschrieben werden, weil eine einzelne Anforderung zum Schreiben von Daten in die Cacheeinheit in mehrere Leseoperationen aufgeteilt werden kann, wenn die maximale Übertragungsgröße für Daten an die Zielplatte kleiner ist als die Anforderungsgröße.

Promote Read Bytes Xfer

Die Gesamtanzahl der Bytes, die von der Zieleinheit gelesen werden, wenn Daten in die Cacheeinheit geschrieben werden.

Anmeldennamen, System-IDs und Kennwörter

Das Betriebssystem muss wissen, wer Sie sind, um Ihnen die richtige Umgebung bereitstellen zu können.

Für die Identifizierung beim Betriebssystem melden Sie sich an. Sie geben Ihren *Anmeldennamen* (auch Benutzer-ID oder Benutzername genannt) und ein *Kennwort* ein. Kennwörter sind eine Form der Sicherheit. Personen, die Ihren Anmeldennamen kennen, können sich nur an Ihrem System anzumelden, wenn ihnen Ihr Kennwort bekannt ist.

Wenn Ihr System als Mehrbenutzersystem eingerichtet ist, besitzt jeder berechtigte Benutzer im System einen Benutzeraccount, ein Kennwort und einen Anmeldennamen. Das Betriebssystem überwacht die von den einzelnen Benutzern verwendeten Ressourcen. Diese Art der Überwachung wird auch als *Systemabrechnung* (oder Accounting) bezeichnet. Jedem Benutzer wird ein privater Speicherbereich des Systems, ein so genanntes *Dateisystem* zugeordnet. Wenn Sie sich anmelden, scheint das Dateisystem nur Ihre Dateien zu enthalten. Tatsächlich sind aber Tausende anderer Dateien auf dem System vorhanden.

Sie können mehrere gültige Anmeldennamen auf einem System haben. Wenn Sie einen anderen Anmeldennamen verwenden möchten, müssen Sie sich nicht vom System abmelden. Sie haben vielmehr die Möglichkeit, Ihre verschiedenen Anmeldennamen gleichzeitig in unterschiedlichen Shells oder nacheinander in einer Shell zu verwenden. Falls Ihr System Teil eines Netzwerks mit Verbindungen zu anderen Systemen ist, können Sie sich auf jedem dieser anderen Systeme anmelden, auf denen Sie einen Anmeldennamen besitzen. Dieser Vorgang wird als *Fernanmeldung* bezeichnet.

Nachdem Sie Ihre Arbeit mit dem Betriebssystem abgeschlossen haben, melden Sie sich ab, um sicherzustellen, dass Ihre Dateien und Daten geschützt sind.

Am Betriebssystem anmelden

Zur Verwendung des Betriebssystems muss das System gestartet, und Sie müssen angemeldet sein. Wenn Sie sich am Betriebssystem anmelden, identifizieren Sie sich damit beim System und erlauben dem System, Ihre Betriebsumgebung einzurichten.

Ihr System kann so eingerichtet sein, dass Sie sich nur zu bestimmten Tageszeiten und an bestimmten Wochentagen anmelden können. Wenn Sie versuchen, sich zu anderen als den zulässigen Zeiten anzumelden, wird der Zugriff verweigert. Ihr Systemadministrator kann Ihre Anmeldezeiten überprüfen.

Sie melden sich in einem Anmeldedialog an. Wenn Sie sich am Betriebssystem anmelden, werden Sie automatisch in Ihr Ausgangsverzeichnis (auch *Anmeldeverzeichnis* oder *Home-Verzeichnis*) gesetzt.

Nachdem Sie Ihr System eingeschaltet haben, melden Sie sich am System an, um eine Sitzung zu starten.

1. Geben Sie an der Eingabeaufforderung **Anmelden:** Ihren Anmeldenamen ein:

Anmelden: *AnmeldeName*

Ist Ihr AnmeldeName beispielsweise *denise*, geben Sie Folgendes ein:

Anmelden: *denise*

2. Wenn die Eingabeaufforderung **Kennwort:** erscheint, geben Sie Ihr Kennwort ein. (Das Kennwort wird bei der Eingabe nicht am Bildschirm angezeigt.)

Kennwort: [Ihr Kennwort]

Falls die Eingabeaufforderung für das Kennwort nicht angezeigt wird, ist kein Kennwort definiert. Sie können mit Ihrer Arbeit am Betriebssystem beginnen.

Falls Ihre Maschine noch nicht eingeschaltet ist, führen Sie vor der Anmeldung folgende Schritte aus:

1. Schalten Sie alle angeschlossenen Einheiten ein.
2. Schalten Sie die Systemeinheit ein. Schieben Sie dazu den Netzschalter der Systemeinheit in die Position **I**.
3. Beobachten Sie die dreistellige Anzeige. Werden die Selbsttests fehlerfrei durchgeführt, bleibt die dreistellige Anzeige leer.

Falls ein Fehler auftritt, der Ihr Eingreifen erfordert, wird ein dreistelliger Code angezeigt, und die Systemeinheit wird gestoppt. Informationen zu den Fehlercodes und zur Fehlerbehebung erhalten Sie von Ihrem Systemadministrator.

Nach erfolgreicher Durchführung der Selbsttests wird ein Anmeldedialog wie der folgende angezeigt:

Anmelden:

Je nach Konfiguration des Systems wird das System nach der Anmeldung entweder in einer Befehlszeilenschnittstelle (Shell) oder einer Grafikschnittstelle (z. B. AIXwindows oder Common Desktop Environment (CDE)) gestartet.

Sollten Sie Fragen bezüglich der Konfiguration Ihres Kennworts oder Benutzernamens haben, wenden Sie sich an Ihren Systemadministrator.

Mehrfach anmelden (Befehl login)

Falls Sie an mehreren Projekten arbeiten und unterschiedliche Accounts für diese Projekte verwalten möchten, können Sie sich mehrfach anmelden. Sie können für die Anmeldung am System denselben oder verschiedene Anmeldenamen verwenden.

Anmerkung: Für jedes System ist eine maximale Anzahl von Anmeldenamen festgelegt, die zu einer Zeit aktiv sein dürfen. Die Anzahl richtet sich nach Ihrer Lizenzvereinbarung und variiert je nach Installation.

Wenn Sie beispielsweise bereits als `denise1` angemeldet sind und Ihr anderer Anmelde-name `denise2` ist, geben Sie an der Eingabeaufforderung den folgenden Befehl ein:

```
login denise2
```

Wenn die Eingabeaufforderung **Kennwort:** erscheint, geben Sie Ihr Kennwort ein. (Das Kennwort wird bei der Eingabe nicht am Bildschirm angezeigt.) Damit sind Sie zweimal am System angemeldet.

Die vollständige Syntax des Befehls **login** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

Benutzer wechseln (Befehl **su**)

Mit dem Befehl **su** (Switch User, Benutzer wechseln) können Sie die Benutzer-ID ändern, die einer Sitzung zugeordnet ist. Sie müssen dazu den Anmeldenamen des Benutzers kennen.

Wenn Sie bereits angemeldet sind und sich als Benutzer-ID `joyce` anmelden möchten, geben Sie an der Eingabeaufforderung folgenden Befehl ein:

```
su joyce
```

Wenn die Eingabeaufforderung **Kennwort:** erscheint, geben Sie das Kennwort für den Benutzer `joyce` ein. Ihre Benutzer-ID ist jetzt `joyce`. Sollten Sie das Kennwort nicht kennen, wird die Anforderung abgelehnt.

Mit dem Befehl **id** können Sie prüfen, ob Sie mit der Benutzer-ID `joyce` angemeldet sind.

Zugehörige Konzepte:

„Benutzer-IDs anzeigen“ auf Seite 627

Wenn Sie die Systemkennungen (IDs) für einen bestimmten Benutzer anzeigen möchten, verwenden Sie den Befehl **id**. Die System-IDs sind Nummern, die Benutzer und Benutzergruppen auf dem System identifizieren.

Zugehörige Informationen:

```
su command syntax
```

Anmeldenachrichten unterdrücken

Nach einer erfolgreichen Anmeldung zeigt der Befehl **login** Tag, Datum und Uhrzeit des letzten erfolgreichen Anmeldeversuchs sowie die Anzahl der nicht erfolgreichen Anmeldeversuche seit der letzten Änderung der Authentifizierungsdaten (in der Regel eine Änderung des Kennworts) für diesen Benutzer an. Sie können diese Nachrichten unterdrücken, indem Sie die Datei `.hushlogin` in Ihrem Ausgangsverzeichnis erstellen.

Geben Sie an der Eingabeaufforderung in Ihrem Ausgangsverzeichnis den folgenden Befehl ein:

```
touch .hushlogin
```

Der Befehl **touch** erstellt eine leere Datei mit dem Namen `.hushlogin`, sofern eine solche nicht vorhanden ist. Bei der nächsten Anmeldung werden alle Anmeldenachrichten unterdrückt. Sie können das System anweisen, nur die Nachrichten desselben Tages aufzubewahren und alle anderen Anmeldenachrichten zu unterdrücken.

Zugehörige Informationen:

```
touch, Befehl
```

Vom Betriebssystem abmelden (Befehle **exit** und **logout**)

Sie können sich auf die folgenden Arten vom Betriebssystem abmelden.

Tastenkombination für Dateiende drücken (Strg-D)

ODER

exit eingeben

ODER

logout eingeben

Nachdem die Abmeldung durchgeführt wurde, erscheint die Eingabeaufforderung **Anmelden:**.

Benutzer-IDs anzeigen

Wenn Sie die Systemkennungen (IDs) für einen bestimmten Benutzer anzeigen möchten, verwenden Sie den Befehl **id**. Die System-IDs sind Nummern, die Benutzer und Benutzergruppen auf dem System identifizieren.

Der Befehl **id** zeigt die folgenden Informationen (sofern anwendbar) an:

- Benutzername und reale Benutzer-ID
- Name der Benutzergruppe und reale Gruppen-ID
- Name der zusätzlichen Gruppen des Benutzers und ggf. zusätzliche Gruppen-IDs

Geben Sie an der Eingabeaufforderung beispielsweise Folgendes ein:

```
id
```

Das System zeigt Informationen ähnlich den folgenden an:

```
uid=1544(sah) gid=300(build) euid=0(root) egid=9(printq) groups=0(system),10(audit)
```

In diesem Beispiel hat der Benutzer den Benutzernamen `sah` mit der ID 1544, den primären Gruppennamen `build` mit der ID 300, den realen Benutzernamen `root` mit der ID 0, den realen Gruppennamen `printq` mit der ID 9 und die beiden zusätzlichen Gruppennamen `system` und `audit` mit den IDs 0 bzw. 10.

Geben Sie an der Eingabeaufforderung beispielsweise Folgendes ein:

```
id denise
```

Das System zeigt Informationen ähnlich den folgenden an:

```
uid=2988(denise) gid=1(staff)
```

In diesem Beispiel hat der Benutzer `denise` die ID 2988 und ist nur der primären Gruppe `staff` mit der ID 1 zugeordnet.

Die vollständige Syntax des Befehls **id** ist in der Veröffentlichung *Commands Reference, Volume 3* beschrieben.

Zugehörige Tasks:

„Benutzer wechseln (Befehl `su`)“ auf Seite 626

Mit dem Befehl **su** (Switch User, Benutzer wechseln) können Sie die Benutzer-ID ändern, die einer Sitzung zugeordnet ist. Sie müssen dazu den Anmeldenamen des Benutzers kennen.

Anmeldenamen anzeigen (Befehle **whoami** und **logname**):

Wenn Sie sich mehrfach anmelden, können Sie leicht den Überblick darüber verlieren, welchen Anmeldenamen Sie gerade verwenden. Mit den Befehlen **whoami** und **logname** können diese Informationen angezeigt werden.

Befehl **whoami** verwenden

Geben Sie an der Eingabeaufforderung den folgenden Befehl ein, um festzustellen, welchen Anmeldenamen Sie gerade verwenden:

```
whoami
```

Das System zeigt Informationen ähnlich den folgenden an:

```
denise
```

In diesem Beispiel ist der verwendete Anmelde-name `denise`.

Die vollständige Syntax des Befehls **whoami** ist in der Veröffentlichung *Commands Reference, Volume 6* beschrieben.

Befehl who am i verwenden

Eine Variante des Befehls **who** ist der Befehl **who am i**. Mit diesem Befehl können Sie den Anmeldenamen, den Terminalnamen und die Uhrzeit der Anmeldung anzeigen. Geben Sie an der Eingabeaufforderung Folgendes ein:

```
who am i
```

Das System zeigt Informationen ähnlich den folgenden an:

```
denise pts/0 Jun 21 07:53
```

In diesem Beispiel ist der Anmelde-name `denise` und der Name des Terminals `pts/0`. Die Anmeldung erfolgte am 21. Juni um 7:53 morgens.

Die vollständige Syntax des Befehls **who** ist in der Veröffentlichung *Commands Reference, Volume 6* beschrieben.

Befehl logname verwenden

Eine Variante des Befehls **who**, der Befehl **logname**, zeigt dieselben Informationen wie der Befehl **who an**.

Geben Sie an der Eingabeaufforderung Folgendes ein:

```
logname
```

Das System zeigt Informationen ähnlich den folgenden an:

```
denise
```

In diesem Beispiel ist der Anmelde-name `denise`.

Namen des Betriebssystems anzeigen (Befehl uname):

Verwenden Sie zum Anzeigen des Betriebssystemnamens den Befehl **uname**.

Geben Sie an der Eingabeaufforderung beispielsweise Folgendes ein:

```
uname
```

Das System zeigt Informationen ähnlich den folgenden an:

```
AIX
```

In diesem Beispiel ist `AIX` der Name des Betriebssystems.

Die vollständige Syntax des Befehls **uname** ist in der Veröffentlichung *Commands Reference, Volume 5* beschrieben.

Systemnamen anzeigen (Befehl `uname`):

Wenn Sie in einem Netz arbeiten und den Namen Ihres Systems anzeigen möchten, verwenden Sie den Befehl `uname` mit dem Flag `-n`. Der Systemname identifiziert Ihr System im Netz. Systemname und Anmelde-ID sind nicht identisch.

Geben Sie an der Eingabeaufforderung beispielsweise Folgendes ein:

```
uname -n
```

Das System zeigt Informationen ähnlich den folgenden an:

```
barnard
```

In diesem Beispiel ist `barnard` der Name des Systems.

Die vollständige Syntax können Sie der Beschreibung des Befehls `uname` in der Veröffentlichung *Commands Reference, Volume 5* entnehmen.

Alle angemeldeten Benutzer anzeigen:

Wenn Sie Informationen zu allen Benutzern anzeigen möchten, die derzeit am lokalen System angemeldet sind, verwenden Sie den Befehl `who`.

Die folgenden Informationen werden angezeigt: Anmelde-name, Systemname sowie Datum und Uhrzeit der Anmeldung.

Anmerkung: Dieser Befehl zeigt nur die auf dem lokalen Knoten angemeldeten Benutzer an.

Geben Sie den folgenden Befehl ein, wenn Sie wissen möchten, wer den lokalen Systemknoten momentan verwendet:

```
who
```

Das System zeigt Informationen ähnlich den folgenden an:

```
joe 1ft/0 Jun 8 08:34  
denise pts/1 Jun 8 07:07
```

Das Beispiel zeigt, dass Benutzer `joe` sich am 8. Juni um 8:34 morgens an Terminal `1ft/0` angemeldet hat.

Weitere Informationen finden Sie in der Beschreibung des Befehls `who` in der Veröffentlichung "Commands Reference".

Kennwörter

Ein eindeutiges Kennwort stellt eine gewisse Systemsicherheit für Ihre Dateien dar.

Ihr System ordnet jedem Account ein Kennwort zu. Die Sicherheit ist für Computersysteme von entscheidender Bedeutung, weil sie verhindert, dass sich nicht autorisierte Personen Zugriff auf das System verschaffen und die Dateien anderer Benutzer manipulieren. Außerdem können über die Sicherheit bestimmten Benutzern exklusive Zugriffsrechte auf bestimmte Befehle und auf bestimmte Dateien erteilt werden. Zum Schutz des Systems gewähren einige Systemadministratoren ihren Benutzern nur den Zugriff auf bestimmte Befehle und Dateien.

Richtlinien für Kennwörter:

Sie müssen ein eindeutiges Kennwort haben. *Kennwörter müssen geheim gehalten werden.* Schützen Sie die Kennwörter so, wie Sie es mit anderen vertraulichen Firmendokumenten tun würden. Beim Festlegen eines Kennworts muss sichergestellt werden, dass es nicht leicht zu erraten, aber auch nicht so kompliziert ist, dass man es notieren muss.

Die Verwendung schwer zu erratender Kennwörter gewährleistet die Sicherheit Ihrer Benutzer-ID. Kennwörter, die persönliche Daten, wie z. B. den eigenen Namen oder Geburtstag enthalten, sind leicht zu erratende Kennwörter. Selbst Wörter aus dem allgemeinen Sprachgebrauch können leicht erraten werden.

Sichere Kennwörter bestehen aus mindestens sechs Zeichen und enthalten auch nicht alphabetische Zeichen. Abwegige Wortkombinationen und Wörter, die absichtlich falsch geschrieben sind, eignen sich ebenfalls gut als Kennwörter.

Anmerkung: Das Kennwort sollte nicht so kompliziert sein, dass es notiert werden muss.

Beachten Sie bei der Auswahl eines Kennworts folgende Richtlinien:

- Verwenden Sie Ihre Benutzer-ID nicht als Kennwort; weder rückwärts geschrieben, verdoppelt noch in anderweitig geänderter Form.
- Verwenden Sie Kennwörter nur einmal. Das System kann so eingerichtet werden, dass die Wiederverwendung von Kennwörtern nicht zugelassen wird.
- Verwenden Sie keine Personennamen als Kennwort.
- Verwenden Sie keine Wörter als Kennwort, die in einem Online-Wörterverzeichnis enthalten sind.
- Verwenden Sie keine Kennwörter mit weniger als sechs Zeichen.
- Verwenden Sie keine Schimpfwörter. Diese gehören zu den Wörtern, die beim Raten eines Kennworts zuerst ausprobiert werden.
- Verwenden Sie Kennwörter, die leicht zu merken sind, damit Sie sie nicht notieren müssen.
- Verwenden Sie Kennwörter, die Zahlen, Klein- und Großbuchstaben enthalten.
- Verwenden Sie als Kennwort eine Kombination aus zwei Wörtern, die durch eine Zahl voneinander getrennt sind.
- Verwenden Sie aussprechbare Kennwörter. Diese lassen sich leichter merken.
- Notieren Sie Ihre Kennwörter nicht. Falls Sie Ihre Kennwörter unbedingt notieren müssen, bewahren Sie sie sicher, beispielsweise in einem abgeschlossenen Schrank auf.

Kennwörter ändern (Befehl `passwd`):

Verwenden Sie zum Ändern Ihres Kennworts den Befehl `passwd`.

1. Geben Sie an der Eingabeaufforderung Folgendes ein:

```
passwd
```

Falls Sie noch kein Kennwort besitzen, überspringen Sie Schritt 2.

2. Die folgende Eingabeaufforderung wird angezeigt:

```
Das Kennwort für Benutzer-ID wird geändert.  
Altes Kennwort von Benutzer-ID:
```

Diese Anforderung verhindert, dass nicht berechtigte Benutzer das Kennwort ändern, während sich der eigentliche Benutzer nicht am System befindet. Geben Sie Ihr aktuelles Kennwort ein, und drücken Sie anschließend die Eingabetaste.

3. Die folgende Eingabeaufforderung erscheint:

```
Neues Kennwort von Benutzer-ID:
```

Geben Sie das gewünschte neue Kennwort ein, und drücken Sie anschließend die Eingabetaste.

4. Es erscheint die folgende Aufforderung zur erneuten Eingabe des neuen Kennworts.

```
Neues Kennwort erneut eingeben:
```

Durch die erneute Eingabe des Kennworts wird verhindert, dass Sie nicht durch einen Tippfehler bei der Eingabe ein Kennwort festlegen, das Sie später nicht mehr rekonstruieren können.

Die vollständige Syntax des Befehls **passwd** ist in der Veröffentlichung *Commands Reference, Volume 4* beschrieben.

Kennwort außer Kraft setzen (Befehl passwd):

Wenn Sie nicht bei jeder Anmeldung Ihr Kennwort eingeben möchten, können Sie Ihr Kennwort außer Kraft setzen.

Geben Sie den folgenden Befehl ein, um Ihr Kennwort außer Kraft zu setzen:

```
passwd
```

Wenn Sie zur Eingabe des neuen Kennworts aufgefordert werden, drücken Sie die Eingabetaste oder die Tastenkombination Strg-D.

Es erfolgt keine weitere Aufforderung zur Eingabe des Kennworts an der Eingabeaufforderung **passwd**. Es wird eine Nachricht angezeigt, die bestätigt, dass das Kennwort außer Kraft gesetzt ist.

Nähere Informationen hierzu und die vollständige Syntax können Sie der Beschreibung des Befehls **passwd** in der Veröffentlichung "Commands Reference" entnehmen.

Befehlsübersicht für Anmeldenamen, System-IDs und Kennwörter

Es sind Befehle für die Bearbeitung von Anmeldenamen, System-IDs und Kennwörtern verfügbar.

Befehle für Anmeldung und Abmeldung

| Eintrag | Beschreibung |
|-----------------|---|
| login | Dieser Befehl leitet die Sitzung ein. |
| logout | Dieser Befehl stoppt alle Prozesse. |
| shutdown | Dieser Befehl beendet den Systembetrieb. |
| su | Dieser Befehl ändert die einer Sitzung zugeordnete Benutzer-ID. |
| touch | Dieser Befehl aktualisiert die Zugriffs- und Änderungszeiten einer Datei und erstellt eine leere Datei. |

Befehle für Benutzer- und Systemidentifikation

| Eintrag | Beschreibung |
|----------------|---|
| id | Dieser Befehl zeigt die System-IDs eines bestimmten Benutzers an. |
| logname | Dieser Befehl zeigt den Anmeldenamen an. |
| uname | Dieser Befehl zeigt den Namen des aktuellen Betriebssystems an. |
| who | Dieser Befehl zeigt die derzeit angemeldeten Benutzer an. |
| whoami | Dieser Befehl zeigt Ihren Anmeldenamen an. |

Befehl für Kennwörter

| Eintrag | Beschreibung |
|----------------|--|
| passwd | Dieser Befehl ändert das Kennwort eines Benutzers. |

Common Desktop Environment

Mit CDE (Common Desktop Environment) können Sie auf Einheiten und Tools im Netz zugreifen, ohne die genaue Position kennen zu müssen. Der Datenaustausch zwischen Anwendungen erfolgt einfach durch Ziehen und Übergeben (Drag and Drop) von Objekten.

Viele Tasks, die früher die Angabe einer komplizierten Befehlszeilensyntax erforderten, können jetzt viel einfacher und in ähnlicher Weise auf verschiedenen Plattformen ausgeführt werden. Beispielsweise können Sie Anwendungen zentral konfigurieren und an Benutzer verteilen. Außerdem können Sie die Sicherheit, Verfügbarkeit und Interoperabilität der Anwendungen für die unterstützten Benutzer zentral verwalten.

Anmerkung: In der Hilfe, der webbasierten Dokumentation und den Hardcopy-Handbüchern zu CDE (Common Desktop Environment) 1.0 wird der Desktop möglicherweise Common Desktop Environment, AIXwindows-Desktop, CDE 1.0 oder einfach nur "der Desktop" genannt.

Automatisches Starten des Desktops aktivieren und inaktivieren

Sie können Ihr System so einstellen, dass Common Desktop Environment automatisch beim Einschalten des Systems gestartet wird.

Diese Konfiguration kann mit System Management Interface Tool (SMIT) oder über die Befehlszeile vorgenommen werden.

Voraussetzungen

Sie müssen Rootberechtigung besitzen, um das automatische Starten des Desktops zu aktivieren oder zu inaktivieren.

Verwenden Sie die folgende Tabelle, um festzustellen, wie der automatische Start des Desktops aktiviert bzw. inaktiviert wird.

Common Desktop Environment automatisch starten und stoppen

| Task | SMIT-Direktaufruf | Befehl oder Datei |
|--|----------------------------|--------------------------------------|
| Automatisches Starten des Desktops aktivieren ¹ | <code>smit dtconfig</code> | <code>/usr/dt/bin/dtconfig -e</code> |
| Automatisches Starten des Desktops inaktivieren ¹ | <code>smit dtconfig</code> | <code>/usr/dt/bin/dtconfig -d</code> |

Anmerkung: Starten Sie das System nach dieser Task erneut.

Common Desktop Environment manuell starten

Verwenden Sie diese Prozedur, um Common Desktop Environment manuell zu starten.

1. Melden Sie sich als Root am System an.
2. Geben Sie in der Befehlszeile Folgendes ein:

```
/usr/dt/bin/dtlogin -daemon
```

Die Anzeige **Desktop-Anmeldung** erscheint. Wenn Sie sich anmelden, wird eine Desktop-Sitzung gestartet.

Common Desktop Environment manuell stoppen

Wenn Sie den Anmeldemanager manuell stoppen, werden alle X-Server und Desktop-Sitzungen, die der Anmeldemanager gestartet hat, gestoppt.

1. Öffnen Sie ein Terminalemulatorfenster, und melden Sie sich als Root an.
2. Geben Sie Folgendes ein, um die Prozess-ID des Anmeldemanagers zu ermitteln:

```
cat /var/dt/Xpid
```

3. Stoppen Sie den Anmeldemanager mit dem folgenden Befehl:

```
kill -term Prozess-ID
```

Desktop-Profil ändern

Wenn Sie sich am Desktop anmelden, wird die Shellumgebungsdatei (`.profile` bzw. `.login`) nicht automatisch gelesen. Der Desktop führt den X-Server aus, bevor Sie sich anmelden. Deshalb muss die von der Datei `.profile` bzw. `.login` bereitgestellte Funktion in diesem Fall vom Anmeldemanager des Desktops zur Verfügung gestellt werden.

Benutzerspezifische Umgebungsvariablen werden in der Datei */Ausgangsverzeichnis/.dtprofile* definiert. Eine Schablone für diese Datei befindet sich in `/usr/dt/config/sys.dtprofile`. Definieren Sie Variablen und Shellbefehle, die nur für den Desktop bestimmt sind, in der Datei `.dtprofile`. Fügen Sie am Ende der Datei `.dtprofile` Zeilen hinzu, um die Shellumgebungsdatei aufzunehmen.

Systemweite (oder globale) Umgebungsvariablen können in den Konfigurationsdateien des Anmeldemanagers definiert werden. Nähere Einzelheiten zum Konfigurieren von Umgebungsvariablen finden Sie in der Veröffentlichung *Common Desktop Environment 1.0: Advanced User's and System Administrator's Guide*.

Bildschirme und Terminals für Common Desktop Environment hinzufügen und entfernen

Sie können Bildschirme und Terminals für Common Desktop Environment hinzufügen und entfernen.

Der Anmeldemanager kann auf einem System mit einer einzigen lokalen Bitmap- oder Grafikkonsole gestartet werden. Es sind allerdings auch andere Szenarien möglich (siehe folgende Abbildung). Sie können Common Desktop Environment von den folgenden Einheiten aus starten:

- lokalen Konsolen,
- fernen Konsole,
- Bitmap- und zeichenorientierten X-Terminal-Systemen, die auf einem Hostsystem im Netz ausgeführt werden.

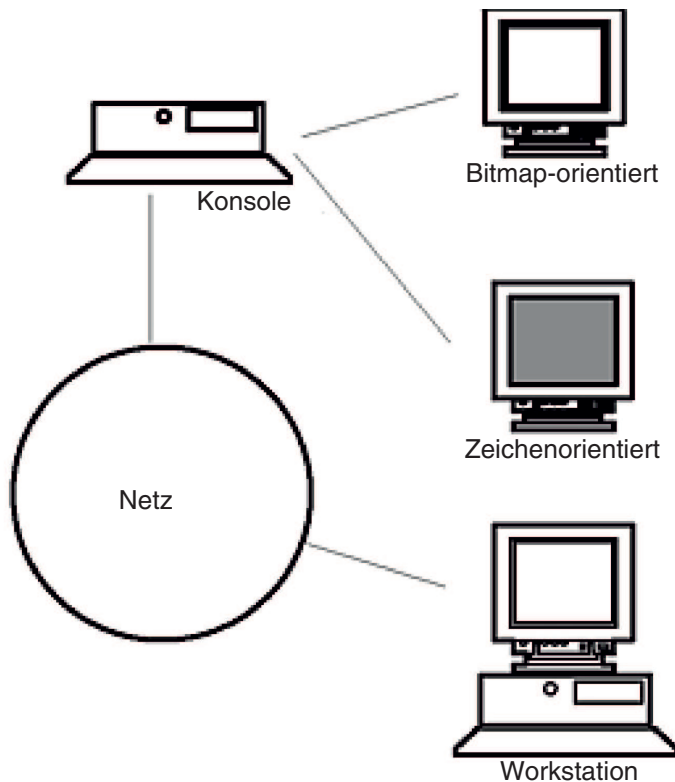


Abbildung 17. Schnittstellenpunkte in CDE. Diese Abbildung veranschaulicht die Verbindungspunkte zwischen einer Konsole, einem Netz, einer grafikfähigen (Bitmap-) Konsole, einer zeichenorientierten Konsole und einer Workstation.

Ein X-Terminal-System setzt sich aus einer Bildschirmeinheit, einer Tastatur und einer Maus zusammen. Auf diesem System wird nur der X-Server ausgeführt. Clients, darunter auch Common Desktop Environment, werden auf einem oder mehreren Hostsystemen in den Netzen ausgeführt. Die Ausgabe der Clients wird an die Bildschirmeinheit des X-Terminals geleitet.

Die folgenden Konfigurationstasks des Anmeldemanagers unterstützen zahlreiche Konfigurationen:

- Lokalen Bildschirm entfernen

- ASCII- bzw. zeichenorientiertes Terminal hinzufügen

Wenn Sie eine Workstation als X-Terminal verwenden möchten, geben Sie in einer Befehlszeile Folgendes ein:

```
/usr/bin/X11/X -query Hostname
```

Der X-Server der Workstation, die als X-Terminal eingesetzt wird, muss folgende Voraussetzungen erfüllen:

- Er muss XDMCP und die Befehlszeilenoption **-query** unterstützen.
- Er muss dem Terminalhost xhost-Berechtigung (in `/etc/X*.hosts`) zuweisen.

Wenn Sie einen lokalen Bildschirm entfernen möchten, müssen Sie den zugehörigen Eintrag in der Datei `Xservers`, die im Verzeichnis `/usr/dt/config` gespeichert ist, entfernen.

Ein *zeichenorientiertes Terminal* bzw. *ASCII-Terminal* ist eine Konfiguration, in der das Terminal keine Bitmap-Einheit ist.

Wenn Sie eine ASCII- bzw. zeichenorientierte Konsole hinzufügen möchten, wenn kein Bitmap-orientierter Bildschirm vorhanden ist, führen Sie die folgenden Schritte aus:

1. Falls die Datei `/etc/dt/config/Xservers` nicht vorhanden ist, kopieren Sie die Datei `/usr/dt/config/Xservers` in das Verzeichnis `/etc/dt/config`.
2. Wenn Sie die Datei `Xservers` nach `/etc/dt/config` kopieren müssen, fügen Sie der Datei `/etc/dt/config/Xconfig` die Zeile `Dtlogin.servers:` hinzu, bzw. ändern Sie sie entsprechend:
`Dtlogin*servers: /etc/dt/config/Xservers`
3. Setzen Sie die Zeile in der Datei `/etc/dt/config/Xservers` auf Kommentar, die den X-Server startet.
`# * Local local@console /path/X :0`

Damit wird das Menü mit den Anmeldeoptionen inaktiviert.

4. Lesen Sie die Konfigurationsdateien des Anmeldemanagers ein.

Wenn Sie eine zeichenorientierte Konsole hinzufügen möchten und ein Bitmap-orientierter Bildschirm vorhanden ist, führen Sie die folgenden Schritte aus:

1. Falls die Datei `/etc/dt/config/Xservers` nicht vorhanden ist, kopieren Sie die Datei `/usr/dt/config/Xservers` in das Verzeichnis `/etc/dt/config`.
2. Wenn Sie die Datei `Xservers` nach `/etc/dt/config` kopieren müssen, fügen Sie der Datei `/etc/dt/config/Xconfig` die Zeile `Dtlogin.servers:` hinzu, bzw. ändern Sie sie entsprechend:
`Dtlogin*servers: /etc/dt/config/Xservers`
3. Editieren Sie in der Datei `/etc/dt/config/Xservers` die Zeile, die den X-Server startet, wie folgt:
`* Local local@none /path/X :0`
4. Lesen Sie die Konfigurationsdateien des Anmeldemanagers ein.

Einheitenanpassung für Common Desktop Environment anzeigen

Sie können den Anmeldemanager von Common Desktop Environment so konfigurieren, dass er auf Systemen mit mehreren Bildschirmen ausgeführt werden kann.

Wenn an ein System mehrere Bildschirme angeschlossen sind, müssen die folgenden Konfigurationsvoraussetzungen erfüllt sein:

- Auf jedem Bildschirm muss ein Server gestartet sein.
- Für jeden Bildschirm muss der Modus ohne Fenster konfiguriert sein.

Möglicherweise ist die Verwendung unterschiedlicher `dtlogin`-Ressourcen für die Bildschirme erforderlich oder empfehlenswert.

Außerdem müssen oder sollten unter Umständen unterschiedliche globale Umgebungsvariablen für die Bildschirme definiert werden.

Server für jede Bildschirmeinheit starten:

Mit der folgenden Prozedur können Sie einen Server für jede Bildschirmeinheit starten.

1. Falls die Datei `/etc/dt/config/Xservers` nicht vorhanden ist, kopieren Sie die Datei `/usr/dt/config/Xservers` in das Verzeichnis `/etc/dt/config`.
2. Wenn Sie die Datei `Xservers` nach `/etc/dt/config` kopieren müssen, ändern Sie die Zeile `Dtlogin.servers:` in der Datei `/etc/dt/config/Xconfig` wie folgt:
`Dtlogin*servers: /etc/dt/config/Xservers`
3. Editieren Sie die Datei `/etc/dt/config/Xservers`, um für jede Bildschirmeinheit einen X-Server zu starten.

Verwenden Sie zum Starten des Servers die folgende allgemeine Syntax:

```
Bildschirmname Bildschirmklasse Bildschirmtyp [ @ite ] Befehl
```

Nur Bildschirme mit zugeordnetem internen Terminalemulator (ITE, Internal Terminal Emulator) unterstützen den *Modus ohne Fenster*. Im Modus ohne Fenster wird der Desktop für den Bildschirm vorübergehend inaktiviert und ein *getty*-Prozess ausgeführt, falls noch kein solcher Prozess gestartet ist. Ein *getty*-Prozess ist ein UNIX-Programm, das den Terminaltyp festlegt und während des Anmeldeprozesses verwendet wird.

Auf diese Weise können Sie sich anmelden und Tasks ausführen, die unter Common Desktop Environment nicht ausgeführt werden können. Wenn Sie sich abmelden, wird der Desktop für den Bildschirm erneut gestartet. Falls auf einer Bildschirmeinheit noch kein *getty*-Prozess ausgeführt wird, startet der Anmeldemanager einen solchen Prozess, wenn der Modus ohne Fenster aktiviert wird.

Wenn `ite` in der Standardkonfiguration nicht angegeben ist, wird dem ITE (`/dev/console`) `display:0` zugeordnet.

Einen anderen Bildschirm als ITE konfigurieren:

Verwenden Sie die folgende Prozedur, um einen anderen Bildschirm als ITE zu konfigurieren.

Gehen Sie wie folgt vor, um einen anderen Bildschirm als ITE anzugeben:

- Setzen Sie in der ITE-Anzeige die Option "ITE" auf die zeichenorientierte Einheit.
- Setzen Sie in allen anderen Anzeigen ITE auf "Keine".

Die folgenden Beispiele zeigen Einträge aus der Datei `Xserver`, die einen Server auf drei lokalen Bildschirmen auf `sysaaa:0` starten. Der Bildschirm `:0` ist die Konsole (ITE).

```
sysaaa:0 local local /usr/bin/X11/X :0
sysaaa:1 local local /usr/bin/X11/X :1
sysaaa:2 local local /usr/bin/X11/X :2
```

Auf Host `sysbbb` ist der Bitmap-Bildschirm `:0` nicht der ITE. Der ITE ist der Einheit `/dev/tty1` zugeordnet. Die folgenden Einträge in der Datei `Xservers` starten Server auf den beiden Bitmap-Bildschirmen, wobei auf `:1` der Modus ohne Fenster aktiviert ist.

```
sysaaa:0 local local@none /usr/bin/X11/X :0
sysaaa:1 local local@tty1 /usr/bin/X11/X :1
```

Bildschirmnamen in Xconfig konfigurieren:

Die reguläre Syntax `hostname:0` für Bildschirmnamen kann in der Datei `/etc/dt/config/Xconfig` nicht verwendet werden.

Gehen Sie wie folgt vor, um den Bildschirmnamen in der Datei Xconfig anzugeben:

- Verwenden Sie ein Unterstreichungszeichen anstelle des Doppelpunkts.
- Verwenden Sie in einem vollständig qualifizierten Hostnamen Unterstreichungszeichen anstelle der Punkte.

Das folgende Beispiel zeigt, wie der Anzeigename in Xconfig konfiguriert wird:

```
Dtlogin.claaa_0.resource: Wert
Dtlogin.sysaaa_prsm_ld_edu_0.resource: Wert
```

Für jeden Bildschirm andere Anmeldemanagerressourcen verwenden:

Gehen Sie wie folgt vor, um für jeden Bildschirm andere Anmeldemanagerressourcen zu verwenden:

1. Falls die Datei /etc/dt/config/Xconfig nicht vorhanden ist, kopieren Sie die Datei /usr/dt/config/Xconfig in das Verzeichnis /etc/dt/config.
2. Öffnen Sie die Datei /etc/dt/config/Xconfig in einem Editor, und geben Sie für jeden Bildschirm eine andere Ressourcendatei an. Beispiel:

```
Dtlogin.DisplayName.resources: Pfad/Datei
```

Pfad steht hier für den Pfadnamen und *Datei* für den Dateinamen der zu verwendenden Xresource-Dateien.

3. Erstellen Sie alle in der Datei Xconfig angegebenen Ressourcendateien. In den Verzeichnissen /usr/dt/config/<Sprache> ist jeweils eine sprachspezifische Xresources-Datei installiert.
4. Legen Sie in jeder dieser Dateien die dtlogin-Ressourcen für den jeweiligen Bildschirm fest.

Das folgende Beispiel enthält Zeilen aus der Datei Xconfig, die unterschiedliche Ressourcendateien für drei Bildschirme angeben:

```
Dtlogin.sysaaa_0.resources: /etc/dt/config/Xresources0
Dtlogin.sysaaa_1.resources: /etc/dt/config/Xresources1
Dtlogin.sysaaa_2.resources: /etc/dt/config/Xresources2
```

Für jeden Bildschirm unterschiedliche Scripts ausführen:

Verwenden Sie diese Prozedur, um ein bestimmtes Script für einen bestimmten Bildschirm auszuführen.

1. Falls die Datei /etc/dt/config/Xconfig nicht vorhanden ist, kopieren Sie die Datei /usr/dt/config/Xconfig in das Verzeichnis /etc/dt/config.
2. Verwenden Sie die Ressourcen startup, reset und setup in der Datei /etc/dt/config/Xconfig, um für jeden Bildschirm andere Scripts anzugeben (diese Dateien werden anstelle der Dateien Xstartup, Xreset und Xsetup ausgeführt):

```
Dtlogin*DisplayName*startup: /Pfad/Datei
Dtlogin*DisplayName*reset: /Pfad/Datei
Dtlogin*DisplayName*setup: /Pfad/Datei
```

Pfad steht hier für den Pfadnamen und *Datei* für den Dateinamen der zu verwendenden Datei. Das Script startup wird nach der Anmeldung des Benutzers und vor dem Start der CDE-Sitzung als Root ausgeführt.

Mit dem Script /usr/dt/config/Xreset können Sie die in der Datei Xstartup festgelegten Einstellungen zurücksetzen. Die Datei Xreset wird beim Abmelden des Benutzers ausgeführt.

Das folgende Beispiel zeigt Zeilen aus der Datei Xconfig, die unterschiedliche Scripts für zwei Bildschirme angeben:

```
Dtlogin.sysaaa_0*startup: /etc/dt/config/Xstartup0
Dtlogin.sysaaa_1*startup: /etc/dt/config/Xstartup1
Dtlogin.sysaaa_0*setup: /etc/dt/config/Xsetup0
Dtlogin.sysaaa_1*setup: /etc/dt/config/Xsetup1
Dtlogin.sysaaa_0*reset: /etc/dt/config/Xreset0
Dtlogin.sysaaa_1*reset: /etc/dt/config/Xreset1
```

Für jeden Bildschirm unterschiedliche globale Umgebungsvariablen festlegen:

Verwenden Sie diese Prozedur, um für jeden Bildschirm systemweite Umgebungsvariablen anzupassen.

Gehen Sie wie folgt vor, um für jeden Bildschirm andere globale Umgebungsvariablen zu definieren:

1. Falls die Datei `/etc/dt/config/Xconfig` nicht vorhanden ist, kopieren Sie die Datei `/usr/dt/config/Xconfig` in das Verzeichnis `/etc/dt/config`.
2. Definieren Sie in der Datei `/etc/dt/config/Xconfig` für jeden Bildschirm gesondert die Ressource `environment`:

```
Dtlogin*DisplayName*environment: Wert
```

Die folgenden Regeln gelten für die Umgebungsvariablen für die Bildschirme:

- Variablenzuordnungen müssen durch ein Leerzeichen oder ein Tabulatorzeichen voneinander getrennt werden.
- Verwenden Sie für das Setzen von **TZ** und **LANG** nicht die Ressource `environment`.
- Shellverarbeitung wird in der Datei `Xconfig` nicht unterstützt.

Das folgende Beispiel zeigt Zeilen aus der Datei `Xconfig`, die Variablen für zwei Bildschirme setzen:

```
Dtlogin*syshere_0*environment:EDITOR=vi SB_DISPLAY_ADDR=0xB00000
Dtlogin*syshere_1*environment:EDITOR=emacs \
SB_DISPLAY_ADDR=0xB00000
```

Drucken und Druckjobs

Für das Drucken in AIX wird eine Vielzahl von Konfigurations- und Setuptools angeboten.

Das Betriebssystem AIX steuert die Darstellung und die Merkmale der finalen Ausgabe abhängig vom verwendeten Drucker. Die Drucker müssen sich nicht in demselben Bereich wie die Systemeinheit und die Systemkonsole befinden. Sie könnten Ihren Drucker beispielsweise direkt an ein lokales System anschließen, oder die bestehende Situation könnte erfordern, dass Druckjobs über ein Netz an ein fernes System gesendet werden.

Damit Druckjobs mit maximaler Effizienz bearbeitet werden, reiht das Betriebssystem AIX jeden Job in eine Warteschlange ein, wo er auf die Verfügbarkeit eines Druckers wartet. Das System kann die Ausgabe einer oder mehrerer Datei(en) in der Warteschlange speichern. Während der Drucker die Ausgabe einer Datei generiert, verarbeitet das System den nächsten Job in der Warteschlange. Dieser Prozess wird so lange fortgesetzt, bis alle Jobs in der Warteschlange gedruckt sind.

Druckjob starten

Verwenden Sie den Befehl `qprt` oder `smit`, um einen Druckjob anzufordern.

- Für lokale Druckjobs muss der Drucker physisch an das System angeschlossen sein. Wenn Sie mit einem Netzdrucker arbeiten, muss dieser angeschlossen und im Netz konfiguriert sein.
- Für ferne Druckjobs muss das System für die Kommunikation mit dem fernen Druckserver konfiguriert sein.
- Damit Sie eine Datei drucken können, müssen Sie *Lesezugriff* auf diese Datei haben. Wenn Sie eine Datei nach dem Drucken löschen möchten, müssen Sie *Schreibzugriff* auf das Verzeichnis haben, in dem die Datei enthalten ist.

Geben Sie die folgenden Informationen an, um einen Druckjob anzufordern:

- Namen der zu druckenden Datei,
- Namen der Druckwarteschlange,
- Anzahl der zu druckenden Kopien,
- ob eine Kopie der Datei auf dem fernen Host erstellt werden soll,
- ob die Datei nach dem Drucken gelöscht werden soll,
- ob eine Benachrichtigung über den Jobstatus gesendet werden soll,
- ob eine Benachrichtigung über den Jobstatus mit dem Mail-Programm des Systems gesendet werden soll,
- Trennstatus,
- Benutzernamen für den Kennsatz "Delivery To" (Adressat),
- Konsolbestätigungsnachricht beim fernen Drucken,
- Dateibestätigungsnachricht beim fernen Drucken,
- Prioritätsebene.

Mit dem Befehl **qprt** können Sie einen Druckjob erstellen und in eine Warteschlange stellen, um die angegebene Datei zu drucken. Werden mehrere Dateien angegeben, werden alle Dateien zu einem Druckjob zusammengefasst. Diese Dateien werden in der Reihenfolge ihrer Eingabe in der Befehlszeile entsprechend gedruckt.

Das Basisformat für den Befehl **qprt** ist folgendermaßen:

```
qprt -PName_der_Warteschlange Dateiname
```

Die folgenden Flags des Befehls **qprt** sind hilfreich:

| Eintrag | Deskriptor |
|---------------------|--|
| -b Anzahl | Gibt den unteren Rand an. Der untere Rand ist die Anzahl von Zeilen, die unten auf jeder Seite leer bleiben sollen. |
| -B Wert | Gibt an, ob Trennseiten (bei Endlospapier sind die Seiten durch eine Perforation getrennt) gedruckt werden sollen. Die Variable <i>Wert</i> steht für eine aus zwei Zeichen bestehende Zeichenfolge. Das erste Zeichen bezieht sich auf das Deckblatt, das zweite Zeichen auf die Abschlusseiten. Die folgenden Zeichen können verwendet werden:
a Das Deckblatt und/oder die Abschlusseite wird für jede Datei in einem Druckjob gedruckt.
n Das Deckblatt und/oder die Abschlusseite wird nicht gedruckt.
g Das Deckblatt und/oder die Abschlusseite wird für jeden Druckjob (Gruppe von Dateien) einmal gedruckt. Das Flag -B ga gibt beispielsweise an, dass am Anfang jedes Druckjobs ein Deckblatt und am Ende jeder Datei in den einzelnen Druckjobs eine Abschlusseite gedruckt werden soll.
Bei fernen Druckumgebungen wird die Standardeinstellung von der fernen Warteschlange auf dem Server festgelegt. |
| -e Option | Gibt an, ob ein verstärkter Druck verwendet werden soll.
+ Gibt an, dass ein verstärkter Druck verwendet werden soll.
! Gibt an, dass kein verstärkter Druck verwendet werden soll. |
| -E Option | Gibt an, ob ein doppelt hoher Druck verwendet werden soll.
+ Gibt an, dass ein doppelt hoher Druck verwendet werden soll.
! Gibt an, dass kein doppelt hoher Druck verwendet werden soll. |
| -f Filtertyp | Eine Kennung (ein Zeichen), die den Filter angibt, mit dem Druckjobs bearbeitet werden müssen, bevor sie an den Drucker gesendet werden. Die verfügbaren Kennungen für Filter sind p (Aufruf des Filters pr) und n (Verarbeitung der Ausgabe des Befehls troff). |
| -i Anzahl | Jede Zeile wird um die angegebene Anzahl von Leerzeichen eingerückt. Die Variable <i>Anzahl</i> muss in der Seitenbreite enthalten sein, die mit dem Flag -w angegeben wird. |

| Eintrag | Deskriptor |
|---|--|
| -K Option | Gibt an, ob ein komprimierter Druck verwendet werden soll.
+ Gibt an, dass der komprimierte Druck verwendet werden soll.
! Gibt an, dass der komprimierte Druck nicht verwendet werden soll. |
| -l Anzahl | Setzt die Seitenlänge auf die angegebene Anzahl von Zeilen. Wird 0 für die Variable <i>Anzahl</i> angegeben, wird die Seitenlänge ignoriert und die Ausgabe als fortlaufende Seite betrachtet. Die Seitenlänge beinhaltet den oberen und den unteren Rand und zeigt die bedruckbare Länge des Papiers an. |
| -L Option | Gibt an, ob bei Zeilen, die die Seitenbreite überschreiten, ein automatischer Zeilenumbruch erfolgen soll, oder ob diese Zeilen am rechten Rand abgeschnitten werden sollen.
+ Gibt an, dass ein automatischer Zeilenumbruch erfolgen soll.
! Gibt an, dass lange Zeilen am rechten Rand abgeschnitten werden sollen. |
| -N Anzahl | Gibt die Anzahl der zu druckenden Kopien an. Ohne dieses Flag wird nur eine Kopie gedruckt. |
| -p Anzahl | Setzt die Zeichendichte auf die angegebene <i>Anzahl</i> Zeichen pro Zoll. Typische Werte für <i>Anzahl</i> sind 10 und 12. Die tatsächlich gedruckte Zeichendichte wird auch durch die für die Flags -K (komprimiert) und -W (doppelte Höhe) angegebenen Werte beeinflusst. |
| -P Warteschlange[:Warteschlangeneinheit] | Gibt den Namen der Druckwarteschlange und den optionalen Namen der Warteschlangeneinheit an. Wird dieses Flag nicht angegeben, wird der Standarddrucker verwendet. |
| -Q Wert | Gibt das Papierformat für den Druckjob an. Der <i>Wert</i> für das Papierformat ist abhängig vom Drucker. Typische Werte sind 1 für Letter-Format und 2 für Legal-Format usw. Weitere Informationen zu den Werten, die bestimmten Papierformaten zugeordnet sind, sind im Druckerhandbuch enthalten. |
| -t Anzahl | Gibt den oberen Rand an. Der obere Rand ist die Anzahl von Zeilen, die oben auf jeder Seite leer bleiben sollen. |
| -w Anzahl | Setzt die Seitenbreite auf die mit der Variablen <i>Anzahl</i> angegebene Anzahl von Zeichen. Die Seitenbreite muss die Anzahl von eingerückten Leerzeichen enthalten, die mit dem Flag -i angegeben werden. |
| -W Option | Gibt an, ob doppelt breiter Druck verwendet werden soll.
+ Gibt an, dass der doppelt breite Druck verwendet werden soll.
! Gibt an, dass der doppelt breite Druck nicht verwendet werden soll. |
| -z Wert | Dreht die Ausgabe von Seitendruckern um die Anzahl von Vierteldrehungen im Uhrzeigersinn, die in der Variablen <i>Wert</i> angegeben wird. Die Werte für die Länge (-l) und Breite (-w) werden automatisch angepasst.

0 Hochformat
1 Querformat rechts
2 Hochformat um 180 Grad gedreht
3 Querformat links |
| -# Wert | Gibt eine spezielle Funktion an.

j Zeigt die Jobnummer für den angegebenen Druckjob an.

h Stellt den Druckjob in die Druckwarteschlange. Der Druckjob hat bis zur erneuten Freigabe den Status ANGEHALTEN .

v Überprüft die angegebenen Werte für das Flag des Drucker-Back-End. Diese Überprüfung ist nützlich, wenn bei der Übergabe eines Druckjobs nach unzulässigen Flag-Werten gesucht wird. Wird die Überprüfung nicht angegeben, kann der Druckjob während der Verarbeitung durch einen falschen Flag-Wert gestoppt werden. |

Die folgende Liste enthält Beispiele für die Verwendung der Flags für den Befehl **qprt**:

- Geben Sie Folgendes ein, um die Datei `meinedatei` unter Verwendung der Standardwerte auf dem ersten verfügbaren Drucker zu drucken, der für die Standarddruckwarteschlange konfiguriert ist:
`qprt minedatei`
- Geben Sie beispielsweise Folgendes ein, um die Datei `meinedatei` mit bestimmten Flag-Werten über eine bestimmte Warteschlange zu drucken und die Flag-Werte bei Übergabe des Druckjobs zu überprüfen:
`qprt -f p -e + -Pfastest -# v minedatei`

Dieser Befehl übergibt die Datei `meinedatei` mit dem Filterbefehl **pr** (Flag **-f p**) und druckt die Datei im Hervorhebungsmodus (Flag **-e +**) auf dem ersten verfügbaren Drucker, der für die Warteschlange **fastest** (Flag **-Pfastest**) konfiguriert ist.

- Geben Sie Folgendes ein, um die Datei `meindedatei` auf Papier mit Legal-Format zu drucken:
`qprt -Q2 minedatei`
- Geben Sie Folgendes ein, um jeweils drei Kopien der Dateien `new.index.c`, `print.index.c` und `more.c` in der Druckwarteschlange `Msp1` zu drucken:
`qprt -PMsp1 -N 3 new.index.c print.index.c more.c`
- Geben Sie Folgendes ein, um die drei Dateien `new.index.c`, `print.index.c` und `more.c` zusammenhängend in dreifacher Kopie zu drucken:
`cat new.index.c print.index.c more.c | qprt -PMsp1 -N 3`

Anmerkung: Das Basisbetriebssystem unterstützt auch den BSD-UNIX-Druckbefehl (**lpr**) und System V-UNIX-Druckbefehl (**lp**). Die Beschreibung der vollständigen Syntax finden Sie in den Abschnitten zu den Befehlen **lpr** und **lp** in der Veröffentlichung *Commands Reference, Volume 3*.

Eine vollständige Beschreibung der Syntax finden Sie im Abschnitt zum Befehl **qprt** in der Veröffentlichung *Commands Reference, Volume 4*.

Sie können einen Druckjob auch mit SMIT anfordern. Geben Sie Folgendes ein, um einen Druckjob mit SMIT zu starten:

```
smit qprt
```

Druckjob abbrechen (Befehl **qcan**)

Sie können den Befehl **qcan** verwenden, um einen Druckjob abzubrechen.

- Für lokale Druckjobs muss der Drucker physisch an das System angeschlossen sein. Wenn Sie mit einem Netzdrucker arbeiten, muss dieser angeschlossen und im Netz konfiguriert sein.
- Für ferne Druckjobs muss das System für die Kommunikation mit dem fernen Druckserver konfiguriert sein.

Beim Abbrechen eines Druckjobs werden Sie aufgefordert, den Namen der Druckwarteschlange, in der sich der Job befindet, und die Nummer des abzubrechenden Jobs einzugeben.

Diese Vorgehensweise gilt sowohl für lokale als auch für ferne Jobs.

Mit dem Befehl **qcan** können Sie einen bestimmten Job in einer lokalen oder fernen Druckwarteschlange oder alle Jobs in einer lokalen Druckwarteschlange abbrechen. Verwenden Sie den Befehl **qchk**, um die Jobnummer zu bestimmen.

Das Basisformat für den Befehl **qcan** ist wie folgt:

```
qcan -P Warteschlangenname -x Jobnummer
```

Eine vollständige Beschreibung der Syntax finden Sie im Abschnitt zum Befehl **qcan** in der Veröffentlichung *Commands Reference, Volume 4*.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **qcan**:

- Geben Sie Folgendes ein, um den Job mit der Jobnummer **123** auf dem Drucker abzurechnen, auf dem der Job ausgeführt wird: `qcan -x 123`
- Geben Sie Folgendes ein, um alle Jobs in der Warteschlange auf dem Drucker **lp0** abzurechnen: `qcan -X -P1p0`

Anmerkung: Das Basisbetriebssystem unterstützt auch den BSD-UNIX-Befehl zum Abbrechen von Druckjobs (**lprm**) und den System V-UNIX-Befehl zum Abbrechen von Druckjobs (**cancel**). Die vollständige Syntax dieser Befehle ist in den Abschnitten **lprm** und **cancel** in der Veröffentlichung in der Veröffentlichung *Commands Reference* beschrieben.

Druckjob abbrechen (SMIT)

Sie können SMIT verwenden, um einen Druckjob abzubrechen.

- Für lokale Druckjobs muss der Drucker physisch an das System angeschlossen sein. Wenn Sie mit einem Netzdrucker arbeiten, muss dieser angeschlossen und im Netz konfiguriert sein.
- Für ferne Druckjobs muss das System für die Kommunikation mit dem fernen Druckserver konfiguriert sein.

Geben Sie Folgendes ein, um mit SMIT einen Druckjob abzubrechen:

```
smit qcan
```

Anschließend können Sie die Druckjobnummer oder den Drucker auswählen.

Einem Druckjob eine Priorität zuweisen (Befehl **qpri**)

Sie können Jobprioritäten mit dem Befehl **qpri** nur in lokalen Warteschlangen zuweisen.

Der Drucker muss physisch an das System angeschlossen sein.

Je höher der Wert ist, desto höher ist die Priorität des Druckjobs. Die Standardpriorität ist **15**. Die maximale Priorität ist für die meisten Benutzer **20** und für Benutzer mit Rootberechtigung und für Mitglieder der Gruppe **printq** (Gruppe **9**) **30**.

Anmerkung: Einem fernen Druckjobs kann keine Priorität zugewiesen werden.

Mit dem Befehl **qpri** kann die Priorität eines bereits übergebenen Druckjobs geändert werden. Benutzer mit Root-Berechtigung und Benutzer der Gruppe **printq** können jedem Job in der Druckwarteschlange eine Priorität zuweisen.

Das Basisformat für den Befehl **qpri** ist folgendermaßen:

```
qpri -# Jobnummer -a Prioritätsstufe
```

Eine vollständige Beschreibung der Syntax finden Sie im Abschnitt zum Befehl **qpri** in der Veröffentlichung *Commands Reference, Volume 4*.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **qpri**:

- Geben Sie Folgendes ein, um die Priorität des Jobs mit der Jobnummer **123** in **18** zu ändern:
`qpri -# 123 -a 18`
- Geben Sie Folgendes ein, um einem lokalen Druckjob bei dessen Übergabe eine Priorität zuzuweisen:
`qpri -PWarteschlangenname -R Prioritätsstufe Dateiname`

Priorität eines Druckjobs ändern (SMIT)

Sie können Jobprioritäten nur in lokalen Warteschlangen zuweisen.

Der Drucker muss physisch an das System angeschlossen sein.

Je höher der Wert ist, desto höher ist die Priorität des Druckjobs. Die Standardpriorität ist **15**. Die maximale Priorität ist für die meisten Benutzer **20** und für Benutzer mit Rootberechtigung und für Mitglieder der Gruppe **printq** (Gruppe 9) **30**.

Anmerkung: Einem fernen Druckjobs kann keine Priorität zugewiesen werden.

Geben Sie Folgendes ein, um die Priorität eines Druckjobs mit SMIT zu ändern:

```
smit qpri
```

Druckjob in eine andere Druckwarteschlange verschieben (Befehl qmov)

Sie können einen Druckjob mit dem Befehl **qmov** aus einer Warteschlange in eine andere verschieben.

Die folgenden Voraussetzungen müssen erfüllt sein, damit diese Task ausgeführt werden kann:

- Der Drucker muss physisch an das System angeschlossen sein.
- Sie müssen Eigner des Druckjobs sein.
- Sie müssen Rootberechtigung haben.
- Sie müssen Mitglied der Gruppe **printq** sein.

Anmerkung: Ein ferner Druckjob kann nicht in eine andere Druckwarteschlange verschoben werden.

Mit dem Befehl **qmov** können Sie einen Druckjob in eine andere Druckwarteschlange verschieben. Sie können einen bestimmten Druckjob oder alle Druckjobs in einer bestimmten Druckwarteschlange verschieben. Sie können auch alle Druckjobs verschieben, die von einem bestimmten Benutzer gesendet werden. Verwenden Sie den Befehl **qchk**, um die Nummer des Druckjobs zu bestimmen. Weitere Informationen finden Sie unter **qchk**.

Das Basisformat des Befehls **qmov** ist folgendes:

```
qmov -mneue_Warteschlange {[ -#Jobnummer ] [ -PWarteschlange ] [ -uBenutzer ]}
```

Sie können einen Druckjob mit einem der folgenden Befehle verschieben:

- `qmov -m Zielwarteschlange -# Jobnummer`
- `qmov -m Zielwarteschlange -P Warteschlange`
- `qmov -m Zielwarteschlange -u Benutzer`

Eine vollständige Beschreibung der Syntax finden Sie im Abschnitt zum Befehl **qmov** in der Veröffentlichung in der Veröffentlichung *Commands Reference*.

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **qmov**:

- Wenn Sie den Job mit der Jobnummer 280 in die Druckwarteschlange hp2 verschieben möchten, geben Sie Folgendes ein:

```
qmov -mhp2 -#280
```

- Geben Sie Folgendes ein, um alle Druckjobs in der Druckwarteschlange hp4D in die Druckwarteschlange hp2 zu verschieben:

```
qmov -mhp2 -Php4D
```

Druckjob in eine andere Druckwarteschlange verschieben (SMIT)

Wenn Ihr Drucker an Ihr System angeschlossen ist, können Sie einen Druckjob mit SMIT in eine andere Druckwarteschlange verschieben.

Wenn Ihr Drucker physisch an Ihr System angeschlossen ist, können Sie einen Druckjob mit System Management Interface Tool (SMIT) in eine andere Druckwarteschlange verschieben.

Die folgenden Voraussetzungen müssen erfüllt sein, damit diese Task ausgeführt werden kann:

- Der Drucker muss physisch an das System angeschlossen sein.
- Sie müssen Eigner des Druckjobs sein.
- Sie müssen Rootberechtigung haben.
- Sie müssen Mitglied der Gruppe **printq** sein.

Anmerkung: Ein ferner Druckjob kann nicht in eine andere Druckwarteschlange verschoben werden.

Geben Sie den folgenden Befehl ein:

```
smit qmov
```

Druckjobs anhalten und freigeben (Befehl qhld)

Sie können Druckjobs mit dem Befehl **qhld** anhalten und freigeben.

Anmerkung: Ferne Druckjobs können nicht angehalten und freigegeben werden.

Die folgenden Voraussetzungen müssen erfüllt sein, damit diese Task ausgeführt werden kann:

- Der Drucker muss physisch an das System angeschlossen sein.
- Sie müssen Eigner des Druckjobs sein.
- Sie müssen Rootberechtigung haben.
- Sie müssen Mitglied der Gruppe **printq** sein.

Mit dem Befehl **qhld** können Sie einen Druckjob anhalten, den Sie an eine Druckwarteschlange gesendet haben. Sie können einen bestimmten Druckjob oder alle Druckjobs in einer bestimmten Druckwarteschlange anhalten. Geben Sie den Befehl **qchk** ein, um die Nummer des Druckjobs zu ermitteln. Das Basisformat des Befehls **qhld** ist folgendes:

```
qhld [ -r ] { [ -#Jobnummer ] [ -PWarteschlange ] [ -uBenutzer ] }
```

Sie können einen Druckjob mit einem der folgenden Befehle anhalten:

- `qhld -#Jobnummer`
- `qhld -PWarteschlange`
- `qhld -uBenutzer`

Sie können einen Druckjob mit einem der folgenden Befehle freigeben:

- `qhld -r -#Jobnummer`
- `qhld -r -PWarteschlange`
- `qhld -r -uBenutzer`

Die folgenden Beispiele veranschaulichen die Verwendung des Befehls **qhld**:

1. Geben Sie beispielsweise den folgenden Befehl ein, um den Job mit der Jobnummer 452 in der Warteschlange, in der sich der Job befindet, anzuhalten:
`qhld -#452`
2. Geben Sie den folgenden Befehl ein, um alle Jobs in der Druckwarteschlange hp2 anzuhalten:
`qhld -Php2`
3. Geben Sie den folgenden Befehl ein, um den Job mit der Jobnummer 452 in der jeweiligen Druckwarteschlange freizugeben:
`qhld -#452 -r`
4. Geben Sie den folgenden Befehl ein, um alle Druckjobs in der Druckwarteschlange hp2 freizugeben:
`qhld -Php2 -r`

Druckjobs anhalten und freigeben (SMIT)

Sie können Druckjobs mit SMIT anhalten und freigeben.

Sie müssen eine der folgenden Personen sein, um einen Druckjob anzuhalten oder freizugeben:

- Eigner des Druckjobs
- Benutzer mit Rootberechtigung
- Mitglied der Gruppe **printq**

Führen Sie den folgenden Befehl aus, um einen Druckjob anzuhalten oder freizugeben:

- `smit qhld`

Status eines Druckjobs überprüfen (Befehl **qchk**)

Sie können den Befehl **qchk** verwenden, um den Status eines Druckjobs zu überprüfen.

- Für lokale Druckjobs muss der Drucker physisch an das System angeschlossen sein. Wenn Sie mit einem Netzdrucker arbeiten, muss dieser angeschlossen und im Netz konfiguriert sein.
- Für ferne Druckjobs muss das System für die Kommunikation mit dem fernen Druckserver konfiguriert sein.

Mit dem Befehl **qchk** können Sie die aktuellen Statusinformationen zu bestimmten Druckjobs, Druckwarteschlangen und Benutzern anzeigen.

Das Basisformat für den Befehl **qchk** ist wie folgt:

```
qchk -P Warteschlangenname -# Jobnummer -u Eigenername
```

Anmerkung: Das Basisbetriebssystem unterstützt auch den BSD-UNIX-Befehl zum Überprüfen der Druckwarteschlange (**lpq**) und den System V-UNIX-Befehl zum Überprüfen der Druckwarteschlange (**lpstat**). Die Beschreibung der vollständigen Syntax finden Sie in den Abschnitten zu den Befehlen **lpq** und **lpstat** in der Veröffentlichung in der Veröffentlichung *Commands Reference*.

Die nachfolgenden Beispiele veranschaulichen die Verwendung des Befehls **qchk**:

- Geben Sie Folgendes ein, um die Standarddruckwarteschlange anzuzeigen:
`qchk -q`
- Geben Sie Folgendes ein, um den ausführlichen Status aller Warteschlangen anzuzeigen, bis diese leer sind, und die Anzeige alle 5 Sekunden zu aktualisieren:
`qchk -A -L -w 5`
- Geben Sie Folgendes ein, um den Status für die Druckwarteschlange **lp0** anzuzeigen:
`qchk -P lp0`
- Geben Sie Folgendes ein, um den Status des Jobs mit der Jobnummer **123** anzuzeigen:
`qchk -# 123`
- Geben Sie Folgendes ein, um den Status aller Jobs in allen Warteschlangen zu prüfen:
`qchk -A`

Statusbedingung für Druckwarteschlangen

Im Folgenden sind verschiedene Statusbedingungen für Druckwarteschlangen aufgelistet:

| Eintrag | Deskriptor |
|----------|--|
| DEV_BUSY | <p>Dieser Status gibt Folgendes an:</p> <ul style="list-style-type: none"> • Es sind mehrere Warteschlangen für eine Druckereinheit (lp0) definiert, und eine andere Warteschlange verwendet momentan die Druckereinheit. • Der Prozess qdaemon hat versucht, die Druckeranschlusseinheit (lp0) zu verwenden, aber eine andere Anwendung verwendet momentan die Druckereinheit. <p>Der Status DEV_BUSY wird erst dann zurückgesetzt, wenn die Warteschlange bzw. Anwendung die Druckereinheit freigibt. Eine andere Möglichkeit ist, den Job bzw. Prozess, der die Druckereinheit gerade belegt, abzubrechen.</p> |
| DEV_WAIT | <p>Gibt an, dass die Warteschlange auf den Drucker wartet, weil der Drucker offline, kein Papier eingelegt, ein Papierstau aufgetreten oder das Kabel nicht richtig angeschlossen, defekt oder falsch verdrahtet ist.</p> <p>Der Status DEV_WAIT wird erst dann zurückgesetzt, wenn der Fehler behoben ist. Für Diagnosetests ist es möglicherweise einfacher, mit dem Befehl enq alle Jobs aus der Warteschlange mit dem Status DEV_WAIT in eine andere Warteschlange zu verschieben, die druckbereit ist oder den Status DOWN hat. Nach der Behebung des Fehlers können Sie alle nicht gedruckten Jobs wieder in die ursprüngliche Warteschlange verschieben.</p> |
| DOWN | <p>Gewöhnlich wird eine Warteschlange in den Status DOWN versetzt, nachdem sie im Status DEV_WAIT war. Dieser Fall tritt ein, wenn der Treiber für die Druckereinheit wegen ungültiger Signale nicht feststellen kann, ob der Drucker erreichbar ist. Einige Drucker können dem Warteschlangensystem möglicherweise nicht signalisieren, dass sie offline sind, und senden deshalb Signale, die anzeigen, dass sie ausgeschaltet sind. Wenn die Druckereinheit signalisiert, dass sie ausgeschaltet ist, wird die Warteschlange in den Status DOWN versetzt.</p> <p>Der Status DOWN wird erst wieder zurückgesetzt, wenn der Fehler, der die Warteschlange inaktiviert hat, behoben ist und der Systemadministrator die Druckwarteschlange wieder aktiviert. Die Druckwarteschlange muss manuell reaktiviert werden, damit sie wieder verwendet werden kann.</p> |
| HELD | Gibt an, dass ein Druckjob angehalten wurde. Der Druckjob kann erst nach der Freigabe vom Spooler verarbeitet werden. |
| QUEUED | Gibt an, dass eine Druckdatei in die Druckwarteschlange eingereicht wurde. |
| READY | Gibt an, dass alle zur Druckwarteschlange gehörenden Komponenten bereit sind, Druckjobs anzunehmen und zu drucken. |
| RUNNING | Gibt an, dass gerade eine Druckdatei gedruckt wird. |

Status eines Druckjobs überprüfen (SMIT)

Sie können den Befehl **smit** verwenden, um den Status eines Druckjobs zu überprüfen.

- Für lokale Druckjobs muss der Drucker physisch an das System angeschlossen sein. Wenn Sie mit einem Netzdrucker arbeiten, muss dieser angeschlossen und im Netz konfiguriert sein.
- Für ferne Druckjobs muss das System für die Kommunikation mit dem fernen Druckserver konfiguriert sein.

Sie können die aktuellen Statusinformationen für bestimmte Jobnummern, Warteschlangen, Drucker und Benutzer anzeigen. Geben Sie Folgendes ein, um mit SMIT den Status eines Druckjobs zu überprüfen:

```
smit qchk
```

Statusbedingung für Druckwarteschlangen

Im Folgenden sind verschiedene Statusbedingungen für Druckwarteschlangen aufgelistet:

| Eintrag | Beschreibung |
|----------|---|
| DEV_BUSY | <p>Dieser Status gibt Folgendes an:</p> <ul style="list-style-type: none"> • Es sind mehrere Warteschlangen für eine Druckereinheit (lp0) definiert, und eine andere Warteschlange verwendet momentan die Druckereinheit. • Der Prozess qdaemon hat versucht, die Druckeranschlusseinheit (lp0) zu verwenden, aber eine andere Anwendung verwendet momentan die Druckereinheit. <p>Der Status DEV_BUSY wird erst dann zurückgesetzt, wenn die Warteschlange bzw. Anwendung die Druckereinheit freigibt. Eine andere Möglichkeit ist, den Job bzw. Prozess, der die Druckereinheit gerade belegt, abzubrechen.</p> |

| Eintrag | Beschreibung |
|----------|---|
| DEV_WAIT | Gibt an, dass die Warteschlange auf den Drucker wartet, weil der Drucker offline, kein Papier eingelegt, ein Papierstau aufgetreten oder das Kabel nicht richtig angeschlossen, defekt oder falsch verdrahtet ist.

Der Status DEV_WAIT wird erst dann zurückgesetzt, wenn der Fehler behoben ist. Für Diagnosetests ist es möglicherweise einfacher, mit dem Befehl enq alle Jobs aus der Warteschlange mit dem Status DEV_WAIT in eine andere Warteschlange zu verschieben, die druckbereit ist oder den Status DOWN hat. Nach der Behebung des Fehlers können Sie alle nicht gedruckten Jobs wieder in die ursprüngliche Warteschlange verschieben. |
| DOWN | Gewöhnlich wird eine Warteschlange in den Status DOWN versetzt, nachdem sie im Status DEV_WAIT war. Dieser Fall tritt ein, wenn der Treiber für die Druckereinheit wegen ungültiger Signale nicht feststellen kann, ob der Drucker erreichbar ist. Einige Drucker können dem Warteschlangensystem möglicherweise nicht signalisieren, dass sie offline sind, und senden deshalb Signale, die anzeigen, dass sie ausgeschaltet sind. Wenn die Druckereinheit signalisiert, dass sie ausgeschaltet ist, wird die Warteschlange in den Status DOWN versetzt.

Der Status DOWN wird erst wieder zurückgesetzt, wenn der Fehler, der die Warteschlange inaktiviert hat, behoben ist und der Systemadministrator die Druckwarteschlange wieder aktiviert. Die Druckwarteschlange muss manuell reaktiviert werden, damit sie wieder verwendet werden kann. |
| HELD | Gibt an, dass ein Druckjob angehalten wurde. Der Druckjob kann erst nach der Freigabe vom Spooler verarbeitet werden. |
| QUEUED | Gibt an, dass eine Druckdatei in die Druckwarteschlange eingereicht wurde. |
| READY | Gibt an, dass alle zur Druckwarteschlange gehörenden Komponenten bereit sind, Druckjobs anzunehmen und zu drucken. |
| RUNNING | Gibt an, dass gerade eine Druckdatei gedruckt wird. |

Dateien zum Drucken formatieren (Befehl pr)

Sie können den Befehl **pr** verwenden, um eine einfache Formatierung von Dateien durchzuführen, die an einen Drucker gesendet werden.

Sie können die Ausgabe des Befehls **pr** zur Formatierung des Textes über eine Pipe an den Befehl **qprt** leiten.

Einige hilfreiche Flags für den Befehl **pr** sind im Folgenden beschrieben:

| Eintrag | Deskriptor |
|-----------------------------|---|
| -d | Verwendet in der Ausgabe doppelte Zeilenvorschübe. |
| -h "Zeichenfolge" | Zeigt die angegebene Zeichenfolge, eingeschlossen in Anführungszeichen (" "), an Stelle des Dateinamens als Seitenkopf an. Trennen Sie das Flag und die Zeichenfolge durch ein Leerzeichen voneinander. |
| -l Zeilen | Überschreibt die Standardeinstellungen von 66 Zeilen pro Seite und setzt die Seitenlänge auf die mit der Variablen <i>Zeilen</i> angegebene Anzahl von Zeilen zurück. Wenn der für <i>Zeilen</i> angegebene Wert kleiner ist als die Summe aus Kopf- und Fußzeilen, werden Kopf- und Fußzeilen unterdrückt (als wäre das Flag -t wirksam). |
| -m | Führt Dateien zusammen. Die Standardausgabe wird wie folgt formatiert. Der Befehl pr schreibt basierend auf der Anzahl der Spaltenpositionen nacheinander jeweils eine Zeile aus jeder angegebenen <i>Datei</i> nebeneinander in Textspalten mit gleicher fester Breite. Verwenden Sie dieses Flag nicht zusammen mit dem Flag -Spalte . |
| -n [Breite][Zeichen] | Führt die Zeilennummerierung basierend auf der mit der Variablen <i>Breite</i> angegebenen Anzahl von Stellen durch. Die Standardeinstellung umfasst 5 Ziffern. Wird die Variable <i>Zeichen</i> (jedes Zeichen, das keine Ziffer ist) angegeben, wird der entsprechende Wert an die Zeilennummer angefügt, um sie von den folgenden Zeichen in der Zeile abzugrenzen. Das Standardtrennzeichen ist das ASCII-Tabulatorzeichen. |
| -o Einrückung | Rückt jede Zeile um die mit der Variablen <i>Einrückung</i> angegebene Anzahl von Zeichenpositionen ein. Die Gesamtanzahl der Zeichenpositionen je Zeile ist die Summe aus <i>Breite</i> und <i>Einrückung</i> . Der Standardwert für die <i>Einrückung</i> ist 0. |
| -s Zeichen | Trennt Spalten durch das mit der Variablen <i>Zeichen</i> angegebene Einzelzeichen und nicht durch die entsprechende Anzahl von Leerzeichen. Der Standardwert für <i>Zeichen</i> ist ein ASCII-Tabulatorzeichen. |
| -t | Unterdrückt die Anzeige der Kopf- und Fußzeilen (jeweils fünf). Der Ausdruck hört bei der letzten Zeile einer Datei ohne Zeilenvorschübe bis zum Ende der Seite auf. |
| -w Breite | Setzt die Anzahl der Spaltenpositionen pro Zeile auf den mit der Variablen <i>Breite</i> angegebenen Wert. Der Standardwert für eine Ausgabe mit mehreren Spalten gleicher Breite ist 72. Sonst gibt es keine weiteren Begrenzungen. Wenn Sie das Flag -s angeben, aber das Flag -w aber nicht, wird standardmäßig eine Breite von 512 Spaltenpositionen verwendet. |

| Eintrag | Deskriptor |
|---------|---|
| -Spalte | Setzt die Anzahl der Spalten auf den mit der Variablen <i>Spalte</i> angegebenen Wert. Der Standardwert ist 1. Verwenden Sie diese Option nicht zusammen mit dem Flag -m . Die Flags -e und -i werden für mehrspaltige Ausgaben verwendet. Textspalten dürfen die Seitenlänge niemals überschreiten (siehe dazu das Flag -l). Wird dieses Flag zusammen mit Flag -t verwendet, ist die Mindestanzahl für die Ausgabe anzugeben. |
| +Seite | Beginnt die Ausgabe bei der angegebenen <i>Seite</i> . Der Standardwert ist 1. |

Eine vollständige Beschreibung der Syntax finden Sie im Abschnitt zum Befehl **pr** in der Veröffentlichung in der Veröffentlichung *Commands Reference*.

Die folgenden Beispiele veranschaulichen die Verwendung der Flags für den Befehl **pr**:

- Wenn Sie eine Datei mit dem Namen `prog.c` mit Überschriften und Seitennummern drucken möchten, geben Sie Folgendes ein:

```
pr prog.c | qprt
```

Dieser Befehl fügt der Datei `prog.c` Seitenüberschriften hinzu und sendet sie an den Befehl **qprt**. Die Überschrift enthält das letzte Änderungsdatum der Datei, den Dateinamen und die Seitennummer.

- Geben Sie Folgendes ein, um einen Titel für die Datei `prog.c` anzugeben:

```
pr -h "MAIN PROGRAM" prog.c | qprt
```

Dieser Befehl druckt die Datei `prog.c` mit dem Titel `MAIN PROGRAM` anstelle des Dateinamens. Das Änderungsdatum und die Seitennummern werden weiterhin gedruckt.

- Geben Sie Folgendes ein, um die Datei `word.lst` mehrspaltig zu drucken:

```
pr -3 word.lst | qprt
```

Dieser Befehl druckt die Datei `word.lst` dreispaltig.

- Geben Sie Folgendes ein, um mehrere Dateien nebeneinander auf dem Papier zu drucken:

```
pr -m -h "Members and Visitors" member.lst visitor.lst | qprt
```

Dieser Befehl druckt die Dateien `member.lst` und `visitor.lst` nebeneinander unter dem Titel **Members and Visitors**.

- Geben Sie Folgendes ein, um die Datei `prog.c` zur späteren Verwendung zu ändern:

```
pr -t -e prog.c > prog.notab.c
```

Dieser Befehl ersetzt Tabulatorzeichen in der Datei `prog.c` durch Leerzeichen und gibt das Ergebnis in der Datei `prog.notab.c` aus. Mit dem Flag **-e** wird der Befehl **pr** angewiesen, die Tabulatorzeichen zu ersetzen. Mit dem Flag **-t** werden die Kopfzeilen unterdrückt.

- Geben Sie Folgendes ein, um die Datei `meinedatei` zweispaltig, im Querformat und in 7-Punkt-Schrift zu drucken:

```
pr -l66 -w172 -2 myfile | qprt -z1 -p7
```

ASCII-Dateien auf einem PostScript-Drucker drucken

Das Text Formatting System verwendet den Filter **enscript**, um ASCII-Druckdateien in PostScript-Dateien zu konvertieren, die auf einem PostScript-Drucker gedruckt werden können.

- Der Drucker muss physisch an das System angeschlossen sein.
- Der Drucker muss konfiguriert und definiert sein.
- Das Konvertierungsprogramm von Text Formatting Services muss installiert sein.

Der Filter **enscript** wird vom Befehl **qprt -da** aufgerufen, wenn ein Druckjob an eine PostScript-Druckwarteschlange übergeben wird. Sie können mit dem Befehl **qprt** folgende Flags angeben, um die Ausgabe bei der Übertragung von ASCII-Dateien an eine PostScript-Druckwarteschlange anzupassen:

| Eintrag | Deskriptor |
|------------------------|---|
| -1+ | Fügt Seitenüberschriften hinzu. |
| -2+ | Formatiert die Ausgabe zweispaltig. |
| -3+ | Druckt die Kopfzeilen, Datumsangaben und Seitennummern in einem einheitlichen Format. Wird manchmal auch als "Hervorhebungsmodus" bezeichnet. |
| -4+ | Druckt die Datei, selbst wenn sie nicht darstellbare Zeichen enthält. |
| -5+ | Listet die Zeichen auf, die in der Schriftart nicht enthalten sind. |
| -h <i>Zeichenfolge</i> | Gibt eine Zeichenfolge an, die für die Kopfzeilen verwendet wird. Wird dieses Flag nicht angegeben, enthält die Kopfzeile den Dateinamen, das Änderungsdatum und die Seitennummer. |
| -l <i>Wert</i> | Gibt die maximale Anzahl von Zeilen pro Seite an. Abhängig von der Punktgröße kann der Ausdruck möglicherweise weniger Zeilen pro Seiten enthalten. |
| -L! | Schneidet Zeilen ab, die die Seitenbreite überschreiten. |
| -p | Gibt die Punktgröße an. Wird dieses Flag nicht angegeben, wird von einer Schriftgröße mit Punktgröße 10 ausgegangen, sofern der zweispaltige gedrehte Modus (-2+ -z1) nicht definiert ist, der mit dem Wert 7 angegeben wird. |
| -s | Gibt den Schriftarttyp an. Wird dieses Flag nicht angegeben, wird die Schriftart Courier verwendet. Der PostScript-Drucker muss Zugriff auf die angegebene Schriftart haben. Die gültigen Werte sind im Folgenden aufgeführt:
Courier-Oblique
Helvetica
Helvetica-Oblique
Helvetica-Narrow
Helvetica-Narrow-Oblique
NewCenturySchlbk-Italic
Optima
Optima-Oblique
Palatino-Roman
Palatino-Italic
Times-Roman
Times-Italic |
| -z1 | Dreht die Ausgabe um 90 Grad (Querformat). |

Die folgende Liste enthält Beispiele für die Verwendung dieser Flags des Befehls **qprt**:

- Geben Sie Folgendes ein, um die ACSII-Datei `myfile.ascii` an den PostScript-Drucker **Msp1** zu senden:
`qprt -da -PMsp1 myfile.ascii`
- Geben Sie Folgendes ein, um die ACSII-Datei `myfile.ascii` an den PostScript-Drucker **Msp1** zu senden und in der Schriftart Helvetica zu drucken:
`qprt -da -PMsp1 -sHelvetica myfile.ascii`
- Geben Sie Folgendes ein, um die ASCII-Datei `myfile.ascii` an den PostScript-Drucker **Msp1** zu senden und in Punktgröße 9 zu drucken:
`qprt -da -PMsp1 -p9 myfile.ascii`

Konvertierung von ASCII in PostScript automatisieren:

Sie können das System so konfigurieren, dass es ASCII-Druckdateien erkennt, die an eine PostScript-Druckwarteschlange übergeben werden, und diese ASCII-Dateien für einen PostScript-Drucker automatisch in PostScript konvertiert.

Viele Anwendungen, die PostScript-Druckdateien generieren, folgen der Konvention und verwenden in PostScript-Dateien die beiden Zeichen `%!` als Anfangszeichen, um die Druckdatei als PostScript-Druckdatei zu kennzeichnen. Führen Sie die folgenden Schritte aus, um das System so zu konfigurieren, dass es ASCII-Druckdateien, die an eine PostScript-Druckdatei gesendet werden, erkennt und diese automatisch in PostScript-Dateien konvertiert, bevor es sie an den PostScript-Drucker sendet:

1. Geben Sie an der Eingabeaufforderung mit `chpq` ein.
2. Geben Sie den Namen der PostScript-Warteschlange ein, oder verwenden Sie das Feature **Liste**, um eine Auswahl in einer Liste mit Warteschlangen zu treffen.
3. Wählen Sie die Menüoption **Druckerkonfiguration** aus.
4. Ändern Sie den Wert des Felds **AUTOMATISCHE ERKENNUNG DES TYPES DER DRUCKDATEI?** in **Ja**.

Jeder der folgenden Befehle konvertiert ASCII-Dateien jetzt in PostScript-Dateien und druckt sie auf einem PostScript-Drucker. Zum Konvertieren der Datei `myfile.ascii` geben Sie einen der folgenden Befehle in der Befehlszeile ein:

```
qprt -Pps myfile.ps myfile.ascii
lpr -Pps myfile.ps myfile.ascii
lp -dps myfile.ps myfile.acsii
```

`ps` ist der Name einer PostScript-Druckwarteschlange.

Automatische Druckdateierkennung außer Kraft setzen:

In manchen Fällen müssen Sie die automatische Bestimmung der Druckdateitypen unter Umständen überschreiben.

Sie können die automatische Bestimmung des Druckdateityps für den PostScript-Druck mit den Flags **-d** und **-s** überschreiben. Das Flag **-d** überschreibt den Standarddruckdateityp, und das Flag **-s** gibt den PostScript-Druck an.

In den folgenden Situationen müssen Sie die automatische Bestimmung des Druckdateityps für den PostScript-Druck möglicherweise überschreiben:

- Geben Sie den folgenden Befehl ein, um eine PostScript-Datei mit dem Namen `myfile.ps` zu drucken, die nicht mit den Zeichen `%!` beginnt:


```
qprt -ds -Pps meineDatei.ps
```
- Geben Sie den folgenden Befehl ein, um die Quellenliste einer PostScript-Datei mit dem Namen `myfile.ps` zu drucken, die mit den Zeichen `%!` beginnt:


```
qprt -da -Pps meineDatei.ps
```

Befehlsübersicht für Druckvorgänge

Es gibt eine Reihe von Befehlen für das Drucken und das Verwalten von Druckwarteschlangen.

| Eintrag | Deskriptor |
|---------------------|---|
| <code>cancel</code> | Bricht Anforderungen an einen Zeilendrucker ab. |
| <code>lp</code> | Sendet Anforderungen an einen Zeilendrucker. |
| <code>lpq</code> | Überprüft die Spooler-Warteschlange. |
| <code>lpr</code> | Reiht Druckjobs in die Warteschlange ein. |
| <code>lprm</code> | Löscht Jobs aus der Spooler-Warteschlange des Zeilendruckers. |
| <code>lpstat</code> | Zeigt Statusinformationen zum Zeilendrucker an. |
| <code>pr</code> | Schreibt eine Datei in die Standardausgabe. |
| <code>qcan</code> | Bricht einen Druckjob ab. |
| <code>qchk</code> | Zeigt den Status einer Druckwarteschlange an. |
| <code>qhld</code> | Hält einen Druckjob an oder gibt einen Druckjob frei. |
| <code>qmov</code> | Verschiebt einen Druckjob in eine andere Druckwarteschlange. |
| <code>qpri</code> | Weist den Jobs in der Druckwarteschlange Prioritäten zu. |
| <code>qprt</code> | Startet einen Druckjob. |

Live Partition Mobility with Host Ethernet Adapters

Mit dem Feature Live Partition Mobility (LPM) with Host Ethernet Adapters (HEA) der IBM PowerVM-Software können Sie eine AIX-LPAR und installierte Anwendungen von einer physischen Partition auf eine andere physische Partition migrieren, während der Host-Ethernet-Adapter (HEA) der Migrationspartition zugeordnet ist.

Während der Migration wird der HEA von der Migrationspartition entfernt und nach Abschluss der Migration nicht auf den Partitionen wiederhergestellt. Die Netzkonnektivität wird dadurch jedoch nicht beeinträchtigt.

Voraussetzungen für Live Partition Mobility with HEA

Bevor Sie LPM with HEA verwenden können, müssen Sie sicherstellen, dass Ihre Systemumgebung die Konfigurations- und Zugriffsvoraussetzungen erfüllt.

Partitionsvoraussetzungen

- Die CEC-Quelle und das CEC-Ziel müssen die Partitionsmigration unterstützen.
- Die AIX-Quellen-LPAR darf keine physischen Ressourcen mit der Einstellung "Required" (Erforderlich) im Profil haben.
- Die AIX-Quellen-LPAR darf neben einem HEA (Host-Ethernet-Adapter) keine weiteren physischen Ressourcen haben.

Zugriffsvoraussetzungen

- Sie müssen Rootberechtigung auf der Partition haben, die Sie migrieren möchten.
- Sie müssen die Berechtigung "hscroot" oder eine äquivalente Berechtigung für die Partitionsmigration in der Quellen-HMC und in der Ziel-HMC haben.

Konfigurationsvoraussetzungen

- Der Host-Ethernet-Adapter darf nicht die Einstellung "Required" (Erforderlich) im Partitionsprofil haben, kann aber die Einstellung "Desired" (Gewünscht) im Profil haben.
- Alle Host-Ethernet-Adapter müssen unter EtherChannel als Primäradapter konfiguriert sein.
- Alle Primäradapter in EtherChannel müssen Host-Ethernet-Adapter sein.
- Der Backup-Adapter für EtherChannel muss ein VEA (Virtual Ethernet Adapter) sein.
- Es muss mindestens ein EtherChannel mit Host-Ethernet-Adapter als Primäradapter und einem Virtual Ethernet Adapter als Backup-Adapter konfiguriert sein.
- Es werden maximal vier EtherChannel unterstützt.
- Das EtherChannel-Failover muss funktionieren.
- Sie müssen sicherstellen, dass Quellensystem und Zielsystem für die Migrationspartition konfiguriert sind.
- Wenn Sie eine Migration zwischen zwei HMCs durchführen, müssen Sie die SSH-Authentifizierung zwischen der Quellen-HMC und der fernen HMC konfigurieren. Sie müssen den Befehl **mkauthkeys** in der Quellen-HMC ausführen, bevor Sie die Migration starten.

Live Partition Mobility with HEA ausführen

Sie können LPM (Live Partition Mobility) with HEA über die SMIT-Schnittstelle ausführen.

Lesen Sie den Abschnitt „Voraussetzungen für Live Partition Mobility with HEA“, bevor Sie versuchen, LPM with HEA zu verwenden.

Gehen Sie wie folgt vor, um eine Partitionsmigration mit LPM with HEA durchzuführen:

1. Geben Sie an der Eingabeaufforderung den SMIT-Direktaufruf **smitty migration** ein, um das Menü von Live Partition Mobility with Host Ethernet Adapter (HEA) anzuzeigen.
2. Geben Sie den Hostnamen oder die IP-Adresse der Quellen-HMC (Hardware Management Console) an.

3. Geben Sie den Benutzernamen für die Quellen-HMC an.
4. Geben Sie **no** ein, wenn das Quellensystem und das Zielsystem von derselben Hardware Management Console verwaltet werden, und fahren Sie dann mit Schritt 5 fort. Geben Sie **yes** ein, wenn das Quellensystem und das Zielsystem von unterschiedlichen HMCs verwaltet werden, und führen Sie anschließend die folgenden Schritte aus:

Anmerkung: Sie müssen den Befehl **mkauthkeys** in der Quellen-HMC ausführen, bevor Sie **yes** eingeben.

- a. Geben Sie den Hostnamen oder die IP-Adresse der fernen HMC (Hardware Management Console) an.
 - b. Geben Sie den Benutzernamen für die Quellen-HMC an.
5. Geben Sie den Namen des Quellensystems an.
 6. Geben Sie den Namen des Zielsystems an.
 7. Geben Sie den Namen der Partition an, die Sie migrieren möchten.
 8. Geben Sie **no** ein, wenn Sie die Migration ohne Validierung durchführen möchten. Geben Sie **yes** ein, wenn Sie nur die Migrationsvalidierung durchführen möchten. Wenn Sie **yes** angeben, wird keine Migration, sondern nur eine Validierung durchgeführt.

Anmerkung: Sie sollten die Validierung der Partitionsmigration vor der eigentlichen Partitionsmigration durchführen.

9. Vergewissern Sie sich, dass alle Felder die richtigen Informationen enthalten, und drücken Sie anschließend die Eingabetaste, um die Migration durchzuführen.

Anmerkung: Sie werden zweimal zur Eingabe des Kennworts aufgefordert. Geben Sie das Kennwort für den Benutzernamen für die Quellen-HMC ein, den Sie zuvor in Schritt 4b angegeben haben.

In diesem Beispiel wird die Partition X von der CEC C, die von HMC A verwaltet wird, auf die CEC D migriert, die von der HMC B verwaltet wird. Führen Sie den Befehl **mkauthkeys** in der HMC A aus, damit die Authentifizierung zwischen HMC A und HMC B durchgeführt wird.

Für diesen Migrationsprozess werden die folgenden Werte in SMIT angegeben:

```
Source HMC Hostname or IP address: A
Source HMC username: hscroot
Migration between two HMCs: yes
  Remote HMC hostname or IP address: B
  Remote HMC username: hscroot
Source system: C
Destination system: D
Migrating Partition name: X
Migration validation only: no
```

Ein weiteres Beispiel ist die Migration von Partition X von der CEC C, die von der HMC A verwaltet wird, auf die CEC D, die von der HMC A verwaltet wird.

Für diesen Migrationsprozess werden die folgenden Werte in SMIT angegeben:

```
Source HMC Hostname or IP address: A
Source HMC username: hscroot
Migration between two HMCs: no
  Remote HMC hostname or IP address:
  Remote HMC username:
Source system: C
Destination system: D
Migrating Partition name: X
Migration validation only: no
```

Wenn ein Migrationsfehler auftritt, folgen Sie der Prozedur zum Ausführen von Live Partition Mobility in der Quellen-HMC.

NIM-Client mit LPM migrieren:

Wenn Live Partition Mobility (LPM) verwendet wird, um eine Maschine von einem physischen Server auf einen anderen zu verschieben, und die Maschine als NIM-Client (Network Installation Management) definiert ist, muss der NIM-Administrator das Attribut *cpuid* für den NIM-Client so aktualisieren, dass der neue Hardwarewert nach Abschluss der LPM-Migration widergespiegelt wird.

Führen Sie die folgenden Schritte aus, um das Attribut *cpuid* zu aktualisieren:

1. Fordern Sie auf dem NIM-Client die neue *cpuid*-ID mit dem folgenden Befehl an:

```
uname -a
```

2. Führen Sie auf dem NIM-Master den folgenden Befehl aus:

```
nim -o change -a cpuid=CPU-ID Client
```

Anmerkung: Das Netzinstallationsprogramm **OS_install** unterstützt die Installation des Betriebssystems Linux nicht mehr, weil die CSM-Unterstützung (Cluster Systems Management) im Betriebssystem AIX entfernt wurde.

Adapter für DLPAR verlagern

Sie müssen den Grafikadapter konfigurieren, bevor Sie einen Adapter für DLPAR-Operationen (Dynamic Logical Partitioning, dynamische logische Partitionierung) verlagern.

Verwenden Sie die folgenden Anweisungen, um einen Grafikadapter, z. B. einen Grafikadapter des Typs FC 5748, dynamisch zu verlagern:

1. Stellen Sie sicher, dass der Grafikadapter (z. B. /dev/lft0) nicht von aktuellen Prozessen (Desktop und Xserver) verwendet wird.
2. Vergewissern Sie sich, dass die Konsole nicht auf lft0 gesetzt ist. Geben Sie den folgenden Befehl ein, um die definierten oder verfügbaren abhängigen Schnittstellen (lft oder rcm) zu ermitteln:

```
lsdev -C | grep lft  
lsdev -C | grep rcm
```

Geben Sie den folgenden Befehl ein, um nach der Definition des Grafikadapters einen übergeordneten PCI-Adapter (Peripheral Component Interconnect) abzurufen:

```
odmget -q name=<cortina adapter name. for instance cor0> CuDv
```

Dieser Befehl stellt Informationen zum übergeordneten Adapter und zum Adapter "cortina" bereit.

3. Geben Sie den folgenden Befehl ein, um alle abhängigen Schnittstellen zu entfernen, um den Adapter für DLPAR-Operationen bereit zu machen:

```
# pdisable lft0  
  
# rmdev -l rcm0  
rcm0 Defined  
  
# rmdev -l lft0  
lft0 Defined  
  
# rmdev -Rdl pci23  
cor0 deleted  
pci23 deleted
```

Die Managementkonsolenschnittstelle kann für DLPAR-Operationen im Grafikadapter verwendet werden.

Loopback-Einheit

Eine Loopback-Einheit ist eine Einheit, die als Blockeinheit für den Zugriff auf Dateien verwendet werden kann.

Die Loopback-Datei kann ein ISO-Image, ein Plattenimage, ein Dateisystem oder das Image eines logischen Datenträgers enthalten. Wenn Sie beispielsweise ein CD-ROM-ISO-Image einer Loopback-Einheit zuordnen und anhängen, können Sie auf das Image auf dieselbe Weise zugreifen wie auf die CD-ROM-Einheit.

Verwenden Sie den Befehl **loopmount**, um eine Loopback-Einheit zu erstellen, um eine bestimmte Datei an die Loopback-Einheit zu binden und um die Loopback-Einheit anzuhängen. Verwenden Sie den Befehl **loopumount**, um eine zuvor an eine Loopback-Einheit angehängte Image-Datei abzuhängen und die Einheit zu entfernen. Die Anzahl der Loopback-Einheiten in AIX ist nicht beschränkt. Standardmäßig wird keine Loopback-Einheit erstellt. Die Einheit muss explizit erstellt werden. Die Blockgröße einer Loopback-Einheit ist immer 512 Bytes.

Eine neue Einheit kann auch mit dem Befehl **mkdev** erstellt werden. Mit dem Befehl **chdev** kann die Einheit geändert und mit dem Befehl **rmdev** entfernt werden. Nachdem Sie eine Einheit erstellt haben, können Sie sie anhängen, um auf das zugrunde liegende Image zuzugreifen, oder Sie können die Einheit als Blockeinheit für unformatierte Ein-/Ausgabe verwenden. Informationen zum zugrunde liegenden Image können mit dem Befehl **ioctl (IOCINFO)** abgerufen werden.

Die folgenden Einschränkungen gelten für Loopback-Einheiten in AIX:

- Der Befehl **varyonvg** für Plattenimages wird nicht unterstützt.
- CD-ISO- und DVD UDF+ISO- sowie andere CD/DVD-Images werden nur im schreibgeschützten Format unterstützt.
- Eine Image-Datei kann nur einer einzigen Loopback-Einheit zugeordnet werden.
- Loopback-Einheiten werden in Workload-Partitionen nicht unterstützt.

Zugehörige Informationen:

loopmount command

loopumount command

ioctl command

AIX Event Infrastructure for AIX and AIX Clusters - AHAFS

AIX Event Infrastructure for AIX and AIX Clusters ist ein Ereignisüberwachungsframework für die Überwachung vordefinierter und benutzerdefinierter Ereignisse.

Einführung in AIX Event Infrastructure

AIX Event Infrastructure ist ein Ereignisüberwachungsframework für die Überwachung vordefinierter und benutzerdefinierter Ereignisse.

In AIX Event Infrastructure ist ein Ereignis als Änderung eines Status oder Werts definiert, die vom Kernel bzw. von einer Kernelerweiterung erkannt werden kann, sobald die Änderung eintritt. Die Ereignisse, die überwacht werden können, werden als Dateien in einem Pseudodateisystem dargestellt. Im Folgenden sind verschiedene Vorteile von AIX Event Infrastructure aufgelistet:

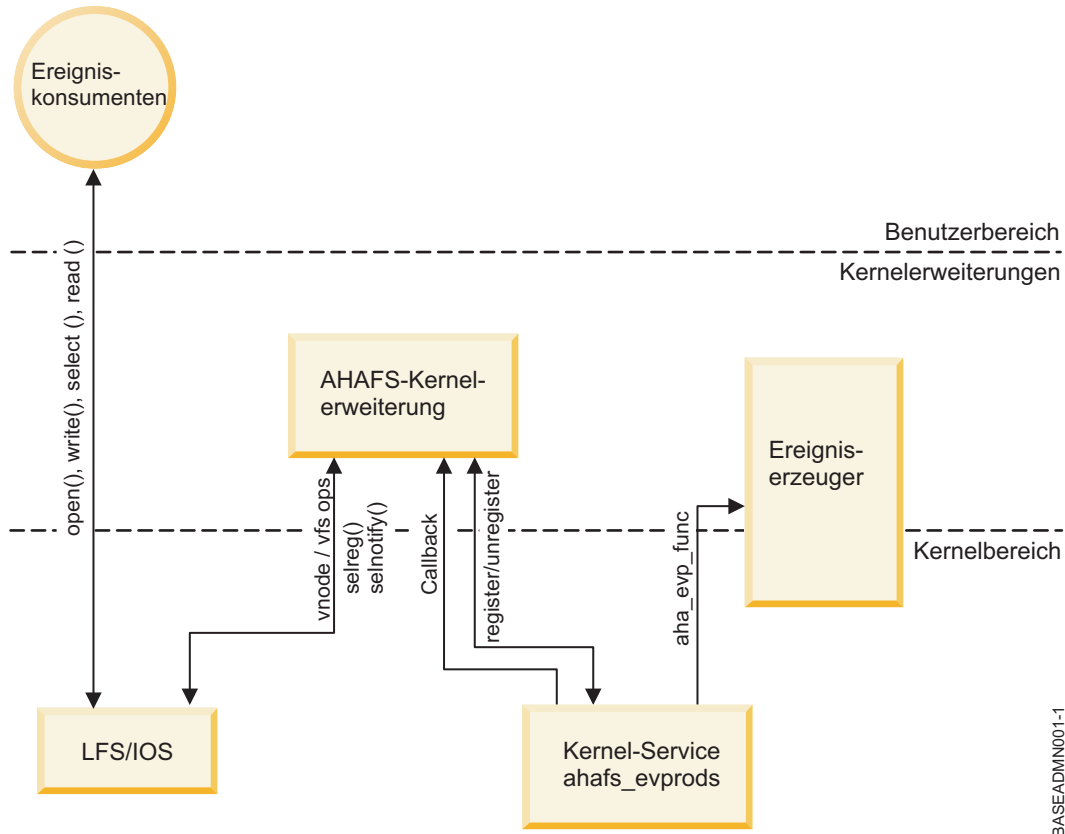
- Eine ständige Abfrage ist nicht erforderlich. Benutzer, die die Ereignisse überwachen, werden benachrichtigt, wenn diese Ereignisse eintreten.
- Detaillierte Informationen zu einem Ereignis (z. B. ein Stack-Trace und Benutzer-/Prozessinformationen) werden dem Benutzer bereitgestellt, der das Ereignis überwacht.
- Es werden vorhandene Dateisystemschnittstellen verwendet, so dass keine neue Anwendungsprogrammierschnittstelle (API) erforderlich ist.

- Die Steuerung wird zu dem Zeitpunkt, zu dem das Ereignis eintritt, an AIX Event Infrastructure übergeben.

Komponenten von AIX Event Infrastructure

AIX Event Infrastructure setzt sich aus den folgenden vier Komponenten zusammen:

- Kernelerweiterung, die das Pseudodateisystem implementiert,
- Ereigniskonsumenten, die die Ereignisse konsumieren,
- Ereigniserzeuger, die Ereignisse erzeugen,
- Kernelkomponente, die als Schnittstelle zwischen der Kernelerweiterung und den Ereigniserzeugern dient.



Kernelerweiterung für AIX Event Infrastructure

Die Kernelerweiterung für AIX Event Infrastructure implementiert ein Pseudodateisystem.

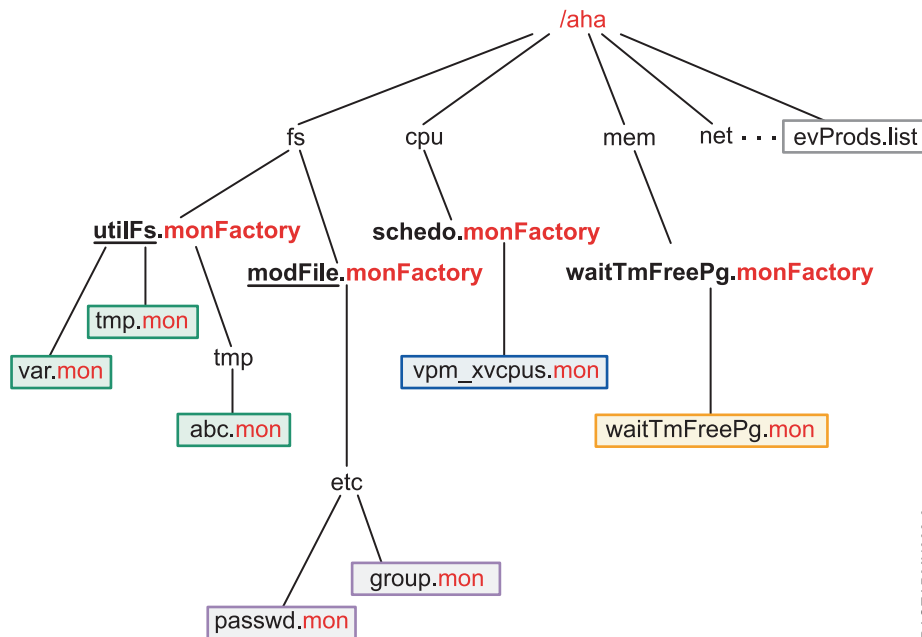
Alle Ereignisse werden als Dateien in diesem Dateisystem dargestellt. Es gibt vier Dateiobjekttypen:

- **Dateien mit der Erweiterung ".list":** Es gibt nur eine einzige Datei mit der Erweiterung **.list** im Pseudodateisystem: **evProds.list**. Diese Datei ist eine Sonderdatei, die die Namen aller derzeit definierten Ereigniserzeuger zurückgibt, wenn sie gelesen wird.
- **Verzeichnisse ".monFactory":** Monitor-Factorys sind ein spezieller Verzeichnistyp. Sie sind Verzeichnisdarstellungen der Ereigniserzeuger. Monitor-Factory-Verzeichnisse und die zugehörigen übergeordneten Unterverzeichnisse werden automatisch für den Benutzer erstellt.
- **Unterverzeichnisse:** Unterverzeichnisse werden sowohl für eine vereinfachte Verwaltung als auch für die Darstellung vollständiger Pfadnamen für Monitordateien (siehe **Dateien mit der Erweiterung ".mon"**) verwendet.
- **Dateien mit der Erweiterung ".mon":** Die Monitordateien stellen Ereignisse dar, die überwacht werden können. Der vollständige Pfadname einer Monitordatei ab der übergeordneten Monitor-Factory, aber

ohne die Erweiterung `.mon` ist die vollständige Darstellung des überwachten Ereignisses. Die Datei `/aha/fs/modFile.monFactory/etc/passwd.mon` wird beispielsweise verwendet, um die Änderungen zu überwachen, die an der Datei `/etc/passwd` vorgenommen werden. Monitordateien können nur unter einer Monitor-Factory existieren.

In diesem Pseudodateisystem können keine anderen regulären Dateien erstellt werden. Da das Dateisystem von AIX Event Infrastructure ein speicherinternes Dateisystem ist, sind maximal 32 KB I-Nodes zulässig. Die Anzahl der verwendeten I-Nodes wird in der Ausgabe des Befehls `df` angezeigt.

Ein Beispiel für das Layout eines Dateisystems von AIX Event Infrastructure sehen Sie im Folgenden:



Anmerkung:

Die Datei `evProds.list` ist direkt unterhalb des Dateisystemstamms vorhanden und enthält die Liste der Ereigniskonsumenten, die definiert sind und in dieser Betriebssysteminstanz verwendet werden können.

Unter Verwendung der LFS-Schnittstelle übersetzt AIX Event Infrastructure die in Monitordateien eingegebenen Texte in Spezifikationen, die angeben, wie der Benutzer über Ereignisvorkommen zu benachrichtigen ist. Sobald ein Benutzer einen Aufruf `select()` oder einen blockierenden Aufruf `read()` absetzt, um den Start der Überwachung anzuzeigen, weist AIX Event Infrastructure den entsprechenden Ereigniskonsumenten an, mit der Überwachung des angegebenen Ereignisses zu beginnen.

Wenn ein Ereignisvorkommen erkannt wird, benachrichtigt AIX Event Infrastructure alle wartenden Konsumenten, deren Überwachungskriterien erfüllt sind.

Ereigniskonsumenten

Ereigniskonsumenten sind Benutzeradressbereichsprozesse, die auf das Eintreten von Ereignissen warten.

Konsumenten konfigurieren die Ereignisüberwachung, indem sie Informationen in eine Monitordatei schreiben, die angeben, wie und wann die Benachrichtigungen erfolgen sollen. Konsumenten können auf Ereignisbenachrichtigungen in einem Aufruf `select()` oder in einem blockierenden Aufruf `read()` warten.

AIX Event Infrastructure ist nicht Thread-sicher. Prozesse sollten nicht mehrere Threads für die Überwachung desselben Ereignisses verwenden.

Ereigniserzeuger

Ereigniserzeuger sind Codeabschnitte im Kernel bzw. in einer Kernelerweiterung, die ein Ereignis erkennen können.

Wenn ein überwachtes Ereignis eintritt, benachrichtigt der Ereigniserzeuger die Kernelerweiterung für AIX Event Infrastructure und sendet zugehörige Informationen zum Ereignis für die Übergabe an den Konsumenten.

Derzeit gibt es zwei Hauptklassen von Ereigniserzeugern:

- Ereigniserzeuger, die Statusänderungen überwachen
- Ereigniserzeuger, die überwachen, ob ein Wert benutzerdefinierte Schwellenwerte überschreitet

Kernel-Service "ahafs_evprods"

Der Kernel-Service **ahafs_evprods** vereinfacht die Kommunikation zwischen der Kernelerweiterung für AIX Event Infrastructure und Ereigniserzeugern.

Zur Vereinfachung der Kommunikation zwischen der Kernelerweiterung für AIX Event Infrastructure und Ereigniserzeugern wird der Kernel-Service **ahafs_evprods** exportiert. Im Kernel wird eine Liste registrierter Ereigniserzeuger verwendet, um Ereigniserzeuger zu suchen und Informationen zwischen entsprechenden Ereigniserzeugern und Kernelerweiterungen zu übergeben.

AIX Event Infrastructure konfigurieren

Erforderliche Schritte für die Konfiguration von AIX Event Infrastructure.

Die einzigen zum Konfigurieren von AIX Event Infrastructure erforderlichen Schritte sind im Folgenden aufgeführt:

1. Dateigruppe **bos.ahafs** installieren.
2. Verzeichnis für den gewünschten Mountpunkt erstellen.
3. Folgenden Befehl ausführen:

```
mount -v ahafs <Mountpunkt> <Mountpunkt>
```

Beispiel

```
mkdir /aha  
mount -v ahafs /aha /aha
```

Beim Anhängen eines Dateisystems von AIX Event Infrastructure wird die Kernelerweiterung automatisch geladen, und alle Monitor-Factorys werden erstellt. Es kann jeweils nur eine einzige Instanz eines Dateisystems von AIX Event Infrastructure angehängt werden. Ein Dateisystem von AIX Event Infrastructure kann über jedes reguläre Verzeichnis angehängt werden, aber es wird empfohlen, das Verzeichnis **/aha** zu verwenden.

Abstrakte Sicht der Funktionsweise von AIX Event Infrastructure

Ein Konsument kann mehrere Ereignisse überwachen, und mehrere Konsumenten können dasselbe Ereignis überwachen. Jeder Konsument kann wertbasierte Ereignisse mit unterschiedlichen Schwellenwerten überwachen. Hierfür verwaltet die Kernelerweiterung für AIX Event Infrastructure eine Liste mit den Informationen jedes Konsumenten, einschließlich:

- angegebener Wartezeit (**WAIT_IN_READ** oder **WAIT_IN_SELECT**),
- Stufe der angeforderten Informationen,
- zu überwachende Schwellenwerte (bei der Überwachung eines Schwellenwertereignisses),
- Puffer für die Speicherung von Informationen zu Ereignisvorkommen.

Ereignisdaten werden prozessbezogen gespeichert, so dass verschiedene Prozesse, die dasselbe Ereignis überwachen, die Ereignisdaten nicht ändern. Wenn ein Konsument Daten aus einer Monitordatei liest, liest er nur seine eigene Kopie der Ereignisdaten.

Typischer Ablauf bei der Überwachung eines Ereignisses

Die Schritte für die Überwachung eines Ereignisses sind in diesem Abschnitt beschrieben.

1. Ein Prozess versucht, eine Monitordatei zu öffnen oder zu erstellen.
2. AIX Event Infrastructure übergibt den Pfadnamen der Monitordatei an den entsprechenden Ereigniserzeuger. Der Ereigniserzeuger prüft, ob die Monitordatei ein gültiges Ereignis darstellt und ob der Prozess Zugriff auf das Ereignis hat, um es überwachen zu können.
3. Der Prozess schreibt Informationen in die Datei, die Folgendes angeben:
 - a. Wartetyp (**WAIT_TYPE=WAIT_IN_READ** oder **WAIT_TYPE=WAIT_IN_SELECT**). Der Standardwartetyp ist **WAIT_IN_SELECT**.
 - b. Wann die Benachrichtigung erfolgen soll. Für Statusänderungsereignisse muss der Benutzer **CHANGED=YES** angeben. Für Schwellenwertereignisse kann der Benutzer je nach Funktionalität des zugeordneten Ereigniserzeugers **THRESH_HI=<Wert>** und/oder **THRESH_LO=<Wert>** angeben. Es gibt keinen Standardwert für diese Spezifikation, und es ist nicht zulässig, **CHANGED=YES** und **THRESH_*=<Wert>** zusammen anzugeben.
4. AIX Event Infrastructure reserviert daraufhin den Vorprozessblock, in dem diese Informationen gespeichert werden, falls noch keiner für diesen Prozess vorhanden ist, und füllt diesen mit den vom Benutzer geschriebenen Informationen.
5. Der Prozess setzt einen Aufruf **select()** oder einen blockierenden Aufruf **read()** für die Monitordatei ab.
6. AIX Event Infrastructure ruft **ahafs_evprods** auf, um zu prüfen, ob die angegebenen Schwellenwerte für dieses spezielle Ereignis gültig sind. Der Ereigniserzeuger **utilFs** lässt beispielsweise keine Werte > 100 % zu. Wenn der Schwellenwert nicht gültig ist, gibt der Aufruf **select()** bzw. **read()** **RC_FROM_EVPROD** zurück, und beim Lesen der Monitordatei wird **EINVAL** zurückgegeben.
7. Für Schwellenwertereignisse wird nur ein einziger Wert für jeden Schwellenwert (**hi** oder **lo**) zur Überwachung an den Ereigniserzeuger gesendet. Bei einem Aufruf **select()** oder einem blockierenden Aufruf **read()** registriert AIX Event Infrastructure diesen neuen Schwellenwert beim Ereigniserzeuger, sofern eine der folgenden Bedingungen zutrifft:
 - a. Wenn kein anderer Prozess dieses Ereignis überwacht, werden die von diesem Konsumenten angegebenen Schwellenwerte beim Ereigniserzeuger registriert.
 - b. Wenn keine anderen Prozesse dieses Ereignis überwachen und der vom Konsumenten angegebene Wert für **THRESH_LO** höher ist als der derzeit überwachte untere Schwellenwert oder der vom Konsumenten angegebene Wert für **THRESH_HI** kleiner ist als der derzeit überwachte obere Schwellenwert, ruft AIX Event Infrastructure den Kernel-Service **ahafs_evprods** auf, um den derzeit überwachten Schwellenwert zu aktualisieren.
8. Nach der Rückkehr vom Kernel-Service **ahafs_evprods** wird (in einigen Fällen) der tatsächliche Wert des Ereignisses zurückgegeben. Sollte der zurückgegebene tatsächliche Wert einen der Schwellenwerte bereits erreicht bzw. überschritten haben, kehrt der Aufruf **read()** bzw. **select()** sofort zurück, und das **RC_FROM_EVPROD**-Ergebnis **EALREADY** wird im Ereignispuffer aufgezeichnet. Der Aufruf **read()** bzw. **select()** gibt 0 zurück.
9. Für Erzeuger von Statusänderungsereignissen wird immer die Funktion **ahafs_evprods** für die Registrierung des Ereignisses aufgerufen.
10. Nach einer erfolgreichen Registrierung konfiguriert AIX Event Infrastructure die Benachrichtigung. Für Konsumenten, die in **select()** warten, wird die Benachrichtigung über **selreg()** konfiguriert. Für Konsumenten, die in einem Aufruf **read()** blockiert sind, wird der Thread mit **e_sleep_thread()** inaktiviert.
11. Wenn ein Ereigniserzeuger erkennt, dass ein Ereignis eingetreten ist, sendet er eine Benachrichtigung an AIX Event Infrastructure, die Informationen zum Ereignis enthält (d. h. Informationen zum Prozess, der das Ereignis ausgelöst hat, aktueller Wert, Rückkehrcode usw.).

12. Während dieses Rückrufs vom Ereigniserzeuger führt AIX Event Infrastructure die folgenden Aktionen aus:
 - a. Der **ahaNode** für das Ereignis wird bestimmt.
 - b. Die Liste der wartenden Konsumenten wird durchsucht, um festzustellen, wessen Schwellenwerte erreicht bzw. überschritten wurden, um diese Konsumenten mit dem Aufruf **selnotify()** oder **e_wakeup()** zu benachrichtigen. Alle Konsumenten, die auf ein Statusänderungsereignis warten, werden benachrichtigt.
13. Sobald der Prozess über das Ereignis benachrichtigt wurde, liest er die Ereignisdaten aus der Monitordatei. Eine Beispielausgabe für ein Ereignis folgt.

Beispielausgabe für den Erzeuger von Statusänderungsereignissen, der angegeben hat, dass ein Stack-Trace erstellt werden soll:

```
BEGIN_EVENT_INFO
TIME_tvsec=1269377315
TIME_tvnsec=955475223
SEQUENCE_NUM=0
PID=2490594
UID=0
UID_LOGIN=0
GID=0
PROG_NAME=cat
RC_FROM_EVPROD=1000
END_EVENT_INFO
```

Beispiel für ein Schwellenwertereignis:

```
BEGIN_EVENT_INFO
TIME_tvsec=1269378095
TIME_tvnsec=959865951
SEQUENCE_NUM=0
CURRENT_VALUE=2
RC_FROM_EVPROD=1000
END_EVENT_INFO
```

Anmerkung: Aufgrund des asynchronen Charakters der Prozessbenachrichtigung kann der zurückgegebene aktuelle Wert veraltet sein, wenn ein Prozess die Monitordatei liest. Benutzer werden benachrichtigt, wenn der Schwellenwert zum ersten Mal erreicht bzw. überschritten wird, aber andere Operationen, die die überwachten Werte ändern können, werden nicht blockiert.

AIX Event Infrastructure verwenden

Alle Verzeichnisse im Dateisystem von AIX Event Infrastructure haben den Zugriffsmodus 1777 und alle Dateien den Zugriffsmodus 0666.

Derzeit haben alle Verzeichnisse im Dateisystem von AIX Event Infrastructure den Modus 1777 und alle Dateien den Modus 0666. Diese Modi können nicht geändert werden, aber der Eigner der Dateien und Verzeichnisse kann geändert werden. Die Zugriffskontrolle für die Überwachung von Ereignissen erfolgt auf der Ebene des Ereigniserzeugers. Die Erstellungs-/Änderungszeiten werden nicht verwaltet und immer als aktuelle Zeit zurückgegeben, wenn **stat 0** für ein Dateiojekt im Pseudodateisystem ausgeführt wird. Jeder Versuch, diese Zeiten zu ändern, erzeugt einen Fehler.

Ereignisse überwachen

Monitordatei erstellen

Für die Überwachung eines Ereignisses muss die Monitordatei für das Ereignis erstellt werden.

Vor der Überwachung eines Ereignisses muss die Monitordatei für das Ereignis erstellt werden. AIX Event Infrastructure unterstützt **open()** mit dem Flag **O_CREAT**. Als Beispiel werden im Folgenden die erforderlichen Schritte für die Überwachung des Dateisystems **/filesys/clj-fs** auf eine Belegung von 90 % beschrieben.

- Die erforderlichen Unterverzeichnisse müssen ebenfalls erstellt werden:
`mkdir /aha/fs/utlFs.monFactory/filesys`
- Öffnen Sie die Datei `/aha/fs/utlFs.monFactory/filesys/clj-fs.mon`.

Bevor eine Monitordatei erstellt werden kann, ruft die Kernelerweiterung für AIX Event Infrastructure den Ereigniserzeuger auf, um festzustellen, ob das angeforderte Ereignis gültig ist und ob der Benutzer über ausreichende Berechtigungen für die Überwachung des angegebenen Ereignisses verfügt. Im Folgenden sind einige typische Fehler aufgeführt, die beim Erstellen oder Öffnen einer Monitordatei zurückgegeben werden können:

Table 71. Rückkehrcodes

| Rückkehrcode | Details |
|--------------|--|
| ENODEV | Es gibt kein Ereignis, das dem angegebenen Pfad entspricht.
Anmerkung: Ein ENODEV-Fehler kann auch zurückgegeben werden, wenn versucht wird, eine vorhandene Monitordatei zu öffnen, das Ereignis aber nicht mehr vorhanden ist. |
| EPERM | Der Benutzer ist für die Überwachung des angegebenen Ereignisses nicht berechtigt. |
| ENOTSUP | Das angegebene Ereignis unterstützt die Überwachung mit AIX Event Infrastructure nicht. |

In die Monitordatei schreiben

Der Konsumentenprozess schreibt in die Monitordatei, um anzugeben, wie und wann er über Ereignisse benachrichtigt werden möchte.

Sobald die Monitordatei erstellt und geöffnet wurde, schreibt der Konsumentenprozess in die Monitordatei, um anzugeben, wie und wann er über Ereignisse benachrichtigt werden möchte. Diese Daten werden in `<Schlüssel>=<Wert>`-Paaren geschrieben, die durch ein Semikolon oder ein Leerzeichen voneinander getrennt werden können. Die gültigen `<Schlüssel>=<Wert>`-Paare sind im Folgenden aufgelistet:

Table 72. Gültige `<Schlüssel>=<Wert>`-Paare

| Schlüssel | Gültige Werte | Details |
|-----------|--|---|
| CHANGED | YES | Gibt an, dass das zu überwachende Ereignis den Typ <code>AHAFS_THRESHOLD_STATE</code> hat und dass der Konsument benachrichtigt werden soll, wenn sich der Status des Ereignisses ändert. |
| THRESH_HI | 64-Bit-Ganzzahl ohne Vorzeichen im Dezimalformat | Dieser Schlüssel gibt den oberen Schwellenwert für das Ereignis an. Sobald das Ereignis diesen Schwellenwert erreicht bzw. diesen überschreitet, wird der Konsument benachrichtigt.
Anmerkung: Obwohl es sich um eine 64-Bit-Ganzzahl handelt, weisen einige Ereigniserzeuger Einschränkungen in Bezug auf die tatsächlich überwachbaren Werte auf. Die gültigen Werte für <code>THRESH_HI</code> für den Ereigniserzeuger <code>utlFs</code> sind beispielsweise 1 bis 100 einschließlich. Die Gültigkeit des Schwellenwerts für den Ereigniserzeuger wird nicht beim Schreiben, sondern vielmehr beim Aufruf von <code>select()</code> bzw. blockierenden Aufruf von <code>read()</code> geprüft. |

Tabelle 72. Gültige <Schlüssel>=<Wert>-Paare (Forts.)

| Schlüssel | Gültige Werte | Details |
|-----------|--|---|
| THRESH_LO | 64-Bit-Ganzzahl ohne Vorzeichen im Dezimalformat | Dieser Schlüssel gibt den unteren Schwellenwert für das Ereignis an. Sobald das Ereignis diesen Schwellenwert erreicht bzw. diesen unterschreitet, wird der Konsument benachrichtigt.
Anmerkung: Obwohl es sich um eine 64-Bit-Ganzzahl handelt, weisen einige Ereigniserzeuger Einschränkungen in Bezug auf die tatsächlich überwachbaren Werte auf. Die gültigen Werte für THRESH_LO für den Ereigniserzeuger utilFs sind beispielsweise 1 bis 100 einschließlich. Die Gültigkeit des Schwellenwerts für den Ereigniserzeuger wird nicht beim Schreiben, sondern vielmehr beim Aufruf von select() bzw. blockierenden Aufruf von read() geprüft. |
| WAIT_TYPE | WAIT_IN_SELECT (Standardwert),
WAIT_IN_READ | Gibt an, wie der Konsument auf das Ereignis wartet. Wenn der Konsument seine Ausführung blockieren möchten, um in einem Aufruf von select() auf ein Ereignis zu warten, muss er WAIT_IN_SELECT angeben. Wenn der Konsument seine Ausführung blockieren möchten, um in einem Aufruf von read() auf ein Ereignis zu warten, muss er WAIT_IN_READ angeben. |
| INFO_LVL | 1, 2 (Standardwert) oder 3 | Gibt an, welche Ereignisdaten im Puffer des Benutzers protokolliert werden sollen. <ul style="list-style-type: none"> • INFO_LVL=1 protokolliert die Zeitmarke des Ereignisses, die Folgenummer, den Rückkehrcode des Ereigniserzeugers, Benutzerinformationen*, Prozessinformationen*, den Programmnamen* und den aktuellen Wert des Ereignisses (sofern zutreffend). • Bei INFO_LVL=2 werden alle Daten der Stufe 1 sowie die Nachricht des Ereigniserzeugers (sofern zutreffend) aufgezeichnet. • Bei INFO_LVL=3 werden alle Daten der Stufe 2 sowie der Stack-Trace des Ereignisses (sofern zutreffend) aufgezeichnet. Anmerkung: Die Benutzerinformationen, die Prozessinformationen, der Programmname und der Stack-Trace sind nur für Ereigniserzeuger verfügbar, die das Flag AHAFS_STKTRACE_AVAILABLE angeben. Nicht alle Ereigniserzeuger übergeben Nachrichten. Welche Informationen verfügbar sind, können Sie in der Dokumentation zum jeweiligen Ereigniserzeuger nachlesen. Beispiele für die Ereignisausgabe finden Sie im Abschnitt „Ereignisdaten lesen“ auf Seite 664. |

Tabelle 72. Gültige <Schlüssel>=<Wert>-Paare (Forts.)

| Schlüssel | Gültige Werte | Details |
|------------|---|---|
| NOTIFY_CNT | -1 (Standardwert) und alle Werte zwischen 1 und 32767 einschließlich. | NOTIFY_CNT gibt an, wie oft das Ereignis eintreten muss, bevor der Prozess benachrichtigt wird. Wenn der Wert "-1" angegeben ist, wird der Konsument bei jedem Eintreten des Ereignisses benachrichtigt, und jedes Ereignisvorkommen wird im Benutzerpuffer aufgezeichnet. Wenn der Konsument einen positiven Wert ungleich null angibt, wird der Konsument so lange blockiert, bis das Ereignis so oft wie angegeben eingetreten ist. Sobald die angegebene Anzahl der Ereignisvorkommen erreicht ist, werden keine weiteren Ereignisse aufgezeichnet, bis der Konsument in einem anderen Aufruf von select() oder read() blockiert wird. Weitere Informationen finden Sie im Abschnitt „Auf Ereignisse warten“ auf Seite 662. |
| CLUSTER | YES | Wenn das System zu einem Cluster gehört und der Cluster aktiv ist, können Konsumenten diesen Schlüssel angeben, um über Vorkommen dieses Ereignisses auf anderen Knoten im Cluster benachrichtigt zu werden. Nicht alle Ereigniserzeuger unterstützen die clusterweite Überwachung. Dieses Feature ist standardmäßig inaktiviert . Weitere Informationen finden Sie im Abschnitt „Clusterereignisse“ auf Seite 683. |
| BUF_SIZE | Eine positive ganze Zahl bis 1048576. | Dieser Schlüssel gibt die Größe des Puffers (in Byte) an, der für die Aufzeichnung von Ereignisdaten verwendet werden soll. Die Standardgröße ist 2048, und die kleinste reservierte Größe ist 1024 Bytes, selbst wenn der Konsument eine kleinere Größe anfordert. |

Das Schreiben von Informationen in die Monitordatei ist nur die Vorbereitung für einen nachfolgenden Aufruf von **select()** oder **read()** (blockierend). Die Überwachung wird erst gestartet, wenn ein Aufruf von **select()** oder **read()** (blockierend) erfolgt.

Wenn Sie beispielsweise das Dateisystem **/fileys/clj-fs** auf das erste Vorkommen einer Belegung von 90 % in einem blockierenden Aufruf von **read()** überwachen möchten, wird die folgende Zeichenfolge in die Datei **/aha/fs/utlFs.monFactory/fileys/clj-fs.mon** geschrieben:

```
WAIT_TYPE=WAIT_IN_READ THRESH_HI=90 NOTIFY_CNT=1
```

Mögliche Rückkehrcodes eines Aufrufs von **write()** für eine Monitordatei:

Tabelle 73. Rückkehrcodes

| Rückkehrcode | Details |
|--------------|---|
| EINVAL | Wenn ein ungültiger Wert für einen der zuvor genannten Schlüssel angegeben wird, schlägt das Schreiben in die Monitordatei mit EINVAL fehl. Sollten die angegebenen Benachrichtigungsparameter (CHANGED oder THRESH_HI/LO) nicht mit den Funktionen des Ereigniserzeugers übereinstimmen, schlägt der Aufruf von "write" ebenfalls mit EINVAL fehl. Gibt der Konsument beispielsweise CHANGED=YES für den Ereigniserzeuger utilFs an (der nur THRESH_HI/LO überwacht), gibt der Aufruf von "write" den Wert EINVAL zurück. Wenn Sie CLUSTER=YES ohne aktiven Cluster angeben, wird ebenfalls der Code EINVAL zurückgegeben. |
| EBUSY | Wenn ein weiterer Thread im Prozess vorhanden ist, der auf das Ereignis wartet, wird beim Schreiben in die Monitordatei durch einen anderen Thread der Code EBUSY zurückgegeben. |
| ESTALE | Die Monitordatei wurde gelöscht. Für die Überwachung dieses Ereignisses muss der Dateideskriptor geschlossen und anschließend mit O_CREAT erneut geöffnet werden. |
| ENOMEM | Es kann kein temporärer Speicher bzw. Speicher für den Ereignispuffer reserviert werden. |
| ENOSPC | Eine Monitordatei kann von maximal 512 Prozessen überwacht werden. Wenn es bereits 512 Prozesse gibt, die diese Datei geöffnet und in diese geschrieben haben, schlägt der Aufruf von "write" mit ENOSPC fehl. |

Auf Ereignisse warten

Überwachungsspezifikationen werden in die Monitordatei geschrieben.

Nachdem die Überwachungsspezifikationen erfolgreich in die Monitordatei geschrieben wurden, wird die Ausführung des Konsumentenprozesses in **select()** oder **read()** blockiert, damit er auf das Eintreten (oder Vorkommen) eines Ereignisses wartet. Konsumenten werden nur dann über eintretende Ereignisse benachrichtigt, wenn sie in **select()** oder **read()** blockiert sind. Es gibt drei Möglichkeiten für den Prozess, von einer **select()**- oder blockierenden **read()**-Operation zurückzukehren:

1. Das Ereignis ist so oft wie angegeben eintreten.
 - Es tritt ein Fall ein, der kein Fehler ist. Konsumenten müssen Ereignisdaten lesen, um festzustellen, wie das Ereignis zu behandeln ist.
2. Es ist ein Problem beim Konfigurieren des Ereignisses in der Kernelerweiterung für AIX Event Infrastructure aufgetreten.

Fehler können auftreten, bevor das Ereignis für die Überwachung beim Ereigniserzeuger registriert wird.

- **read()**
 - Wenn ein anderer Thread in der Leseoperation wartet, scheitert die Leseoperation mit **EBUSY**.
 - Wenn vor dieser Leseoperation keine Schreiboperation durchgeführt wurde, gibt die Leseoperation lediglich 0 mit 0 gelesenen Bytes zurück.
- **select()**

Anmerkung:

Wegen der Implementierung des Systemaufrufs "select" müssen die zugrunde liegenden Dateisystemoperationen **EBADF** zurückgeben, damit **select()** einen Fehler zurückgibt. Deshalb gibt **select()** **EBADF** zurück, sofern eine der folgenden Bedingungen erfüllt ist:

- Ein anderer Thread versucht einen Systemaufruf "select".
- Die Monitordatei wurde gelöscht.

- Es wurde keine Schreiboperation mit der Angabe von Überwachungsspezifikationen durchgeführt.
- Bei der Registrierung beim IOS-Subsystem ist ein Fehler aufgetreten.

In diesen Fällen sind keine Ereignisdaten vorhanden, die gelesen werden könnten.

3. Beim Konfigurieren des Ereignisses im Ereigniserzeuger ist ein Problem aufgetreten.

Bei dem Versuch, das Ereignis beim Ereigniserzeuger zu registrieren, wird ein Eintrag im Puffer aufgezeichnet, den der Konsument lesen kann. Um festzustellen, welcher Fehler aufgetreten ist, muss das in den Ereignisdaten zurückgegebene Ergebnis **RC_FROM_EVPROD** in der Dokumentation des Ereigniserzeugers referenziert werden. Die Ereignisausgabe für diesen Fall enthält nur die Zeitmarke, die Folgenummer und den Rückkehrcode des Ereigniserzeugers, unabhängig von der angegebenen **INFO_LVL**-Einstellung. Ein Beispiel finden Sie unter „Ereignisdaten lesen“ auf Seite 664.

In diesem Fall gibt **select()** den Wert **EBADF** zurück, aber **read()** den Rückkehrcode der zugrunde liegenden Operation **uio_move**.

Wenn der Konsumentenprozess einen **NOTIFY_CNT**-Wert größer als 1 angibt, werden Informationen zu jedem Ereignisvorkommen im Puffer des Konsumenten aufgezeichnet, bis die angeforderte Anzahl an Ereignissen eingetreten ist. Der Konsumentenprozess wird nur aktiviert, wenn das Ereignis so oft wie angegeben eingetreten ist bzw. bis ein Nichtverfügbarkeitsereignis eintritt. Sobald der Konsumentenprozess aktiviert ist, wird das Ereignis nicht mehr überwacht, bis ein Aufruf **select()** oder ein blockierender Aufruf **read()** für die Monitordatei abgesetzt wird.

Wenn ein Konsument den Wert -1 für **NOTIFY_CNT** angibt, wird der Konsumentenprozess nach jedem Vorkommen des Ereignisses aktiviert, und jedes Ereignis, das nach dem ersten erfolgreichen Aufruf von **select()** bzw. einem **read()** (blockierend) eintritt, wird im Konsumentenpuffer aufgezeichnet.

Die Aufrufe **select()** und **read()** werden nicht blockiert, wenn der Puffer noch nicht gelesene Daten enthält.

Vorkommen von Nichtverfügbarkeitsereignissen

Für einige Ereigniserzeuger können Ereignisse eintreten, die bewirken, dass das überwachte Ereignis ungültig wird.

Es folgen einige Beispiele:

- Beendigung eines Prozesses für **processMon** und **pidProcessMon**
- Abhängen eines Dateisystems, das die überwachten Dateien für **modDir** und **modFile** enthält
- Abhängen eines Dateisystems, das von **utilFs** überwacht wird
- Entfernen oder Umbenennen einer Datei, die von **modDir** oder **modFile** überwacht wird
- Entfernen eines Ereigniserzeugers, der derzeit für die Überwachung von Ereignissen verwendet wird (**RC_FROM_EVPROD** ist in diesem Fall **ENODEV**).

Nach dem Auslösen eines Nichtverfügbarkeitsereignisses können die Konsumenten die Überwachung für dieses Ereignisses möglicherweise erst dann fortsetzen, wenn es wieder gültig ist. Beispiele für Ereignisse, die wieder gültig werden:

- Erneutes Anhängen eines überwachten Dateisystems
- Neuerstellung einer überwachten Datei, die gelöscht wurde
- Erneutes Erzeugen eines Prozesses, der überwacht wurde

Wenn ein lokales Nichtverfügbarkeitsereignis ausgelöst wird, entfernt die Kernelerweiterung für AIX Event Infrastructure die betroffenen Monitordateien. Wenn eine Monitordatei gelöscht wird, können Konsumenten, die die Datei noch geöffnet haben, ihre Ereignisdaten zwar lesen, aber weder in diese Monitordatei schreiben noch blockiert werden, um auf das Eintreten eines Ereignisses in dieser Monitordatei zu warten. Wenn ein Nichtverfügbarkeitsereignis vom Konsumenten erkannt wird, müssen Sie die entspre-

chende Aktion einleiten (die voraussichtlich bewirkt, dass das Ereignis wieder gültig wird), den Dateide-skriptor für die Monitordatei schließen und die Monitordatei anschließend mit dem Flag **O_CREAT** er-neut öffnen.

Vorkommen lokaler Nichtverfügbarkeitsereignisse bewirken auch, dass die Blockierung von **select()** und **read()** aufgehoben wird, bevor die angeforderte Anzahl an Ereignisvorkommen erreicht ist, wenn für den Konsumenten **NOTIFY_CNT > 1** angegeben ist. Wenn ein Konsument beispielsweise die Datei **/foo** mit **NOTIFY_CNT=3** überwacht, kehrt der Konsument von **select()** bzw. **read()** zurück, wenn **/foo** entfernt wird, selbst wenn dies das erste Vorkommen eines Ereignisses mit **/foo** ist.

AIX Event Infrastructure für Abfragen verwenden

AIX Event Infrastructure setzt nicht voraus, dass Ereigniserzeuger immer den aktuellen Wert von Ereig-nissen verwalten, die überwacht werden können.

Damit kann eine höhere Leistung erzielt werden, da Ereigniserzeuger keine Aufwände für die Verwal-tung dieses Werts produzieren, wenn Vorkommen des Ereignisses von niemandem überwacht werden.

Dies verursacht jedoch Probleme bei der Verwendung synchroner Abfragen. Da es nicht immer möglich ist, den aktuellen Wert eines Ereignisses zu jeder Zeit abzurufen, werden Aufrufe von **poll()** bzw. syn-chroner Aufrufe von **select()** wie folgt gehandhabt:

- Wenn ein Prozess **select()** oder **poll()** zum ersten Mal für eine Monitordatei absetzt, registriert die Ker-nelerweiterung für AIX Event Infrastructure dieses Ereignis für die Überwachung beim Ereigniserzeu-ger.
 - Für Erzeuger von Schwellenwertereignissen, die den aktuellen Wert verwalten, wird bei der Ereig-nisregistrierung der aktuelle Wert an die Kernelerweiterung für AIX Event Infrastructure zurückge-gaben. Dieser Wert wird zu diesem Zeitpunkt mit dem Schwellenwert des Konsumenten verglichen. Wenn der Schwellenwert des Konsumenten überschritten ist, zeigt **select()** bzw. **poll()** an, dass das Ereignis eingetreten ist, und gibt **RC_FROM_EVPROD** mit dem Wert **EALREADY** zurück.
- **POLLSYNC**-Flags werden ignoriert. Ein Ereignis bleibt so lange beim Ereigniserzeuger registriert, bis das Ereignis den festgelegten Wiederholungszähler erreicht bzw. bis der Benutzer die Datei schließt.
- Nachfolgende Aufrufe von **poll()** weisen das folgende Verhalten auf:
 - Wenn das Ereignis noch nicht eingetreten ist, kehrt der Aufruf ohne Rückgabeereignisse zurück.
 - Wenn die Anzahl der Vorkommen des Ereignisses seit dem letzten Aufruf von **poll()** den angegebe-nen Wert erreicht, zeigen die Rückgabeereignisse an, dass das Ereignis eingetreten ist.

Ereignisdaten lesen

Die Ereignisdaten in AIX Event Infrastructure setzen sich aus Schlüssel/Wert-Paaren zusammen.

Ereignisdaten können nur ein einziges Mal gelesen werden, und es wird nur ein einziges Ereignisvor-kommen mit Daten in einem einzigen Aufruf von **read()** zurückgegeben. Angenommen, zwei Ereignisse sind eingetreten, bevor der Konsument Daten aus der Monitordatei gelesen hat, und jedes Ereignis ent-hält eine Datenmenge von 256 Bytes. Wenn der Konsument **read()** für 4096 Bytes aufruft, werden nur die 256 Bytes des ersten Ereignisses an den Benutzer zurückgegeben. Es muss ein zweiter Aufruf von **read()** abgesetzt werden, um die Daten aus dem zweiten Ereignis abzurufen. Angegebene Offsets werden igno-riert, und die Daten werden ab dem letzten noch nicht gelesenen Byte gelesen.

Ereignisdaten haben eine maximale Größe von 4096 Bytes, obwohl die meisten Ereignisse viel kleiner sein werden (< 512 Bytes). Es wird empfohlen, beim Lesen von Ereignissen einen Puffer zu verwenden, der groß genug ist, um zu verhindern, dass nur ein Teil eines Ereignisses gelesen wird.

Ereignisdaten in AHAFS setzen sich aus **Schlüsselwort = Wert**-Paaren zusammen. Ausgenommen sind **BUF_WRAP**, **EVENT_OVERFLOW**, **BEGIN_EVENT_INFO**, **END_EVENT_INFO**, **BEGIN_EVPROD_IN-FO**, **END_EVPROD_INFO** und **STACK_TRACE**, die spezielle Schlüsselwörter ohne Werte sind. Im Fol-genden sind die Schlüsselwörter aufgelistet, die Sie in Ereignisdaten sehen können:

Tabelle 74. Schlüsselwörter

| Schlüssel | Wert | Details |
|-------------------------|---------|--|
| BUF_WRAP | Ohne | Der Konsumentenpuffer wird wie ein Ringpuffer behandelt. Wenn nicht gelesene Daten mit den aktuellen Ereignisdaten überschrieben werden, ist dieses Schlüsselwort die nächste Zeichenfolge, die vom Aufruf von <code>read()</code> zurückgegeben wird, selbst wenn der Konsument den vorherigen Eintrag nur teilweise gelesen hat. Der nachfolgende Aufruf von <code>read()</code> gibt das nächste vollständige Ereignis zurück. |
| EVENT_OVERFLOW | Ohne | Wenn die Ereignisdaten zu groß für den Ereignisdatenpuffer des Konsumenten sind, wird dieses Schlüsselwort beim ersten Aufruf von <code>read()</code> zurückgegeben. Bei einem nachfolgenden Aufruf von <code>read()</code> werden alle Daten zurückgegeben, die in den Puffer passen.
Anmerkung: Wenn <code>EVENT_OVERFLOW</code> gefunden wird, ist die Endzeichenfolge <code>END_EVENT_INFO</code> nicht vorhanden. |
| BEGIN_EVENT_INFO | Ohne | Dieses Schlüsselwort kennzeichnet den Anfang der Daten für ein Ereignisvorkommen. |
| END_EVENT_INFO | Ohne | Dieses Schlüsselwort kennzeichnet das Ende der Daten für das jeweilige Ereignisvorkommen. |
| TIME_tvsec TIME_tvnssec | Integer | In diesen beiden Feldern wird die Zeitmarke des Ereignisvorkommens in Sekunden und Nanosekunden seit der Epoche aufgezeichnet. |
| SEQUENCE_NUM | Integer | In diesem Feld wird aufgezeichnet, wie oft das Ereignis seit der letzten erfolgreichen Ausführung eines Aufrufs <code>select()</code> oder eines blockierenden Aufrufs <code>read()</code> eingetreten ist. Diese Zahl wird auf 0 zurückgesetzt, wenn der Aufruf von <code>select()</code> bzw. der blockierende Aufruf von <code>read()</code> fehlschlägt oder wenn der Konsument die Überwachung des Ereignisses (durch Überschreiben der Ereignisüberwachungsspezifikationen oder durch Erreichen einer Anzahl von Ereignisvorkommen, die dem angegebenen <code>NOTIFY_CNT</code> -Wert entspricht) einstellt. |
| PID | Integer | Die Prozess-ID des Prozesses, der das Ereignisvorkommen ausgelöst hat. Die ID ist nur bei einem Ereigniserzeuger verfügbar, für den die Funktion <code>AHAFS_STKTRACE_AVAILABLE</code> , aber nicht die Funktion <code>AHAFS_CALLBACK_INTRCNTX</code> angegeben ist. |
| UID | Integer | Die effektive Benutzer-ID des Benutzers, der das Ereignisvorkommen ausgelöst hat. Die ID ist nur bei einem Ereigniserzeuger verfügbar, für den die Funktion <code>AHAFS_STKTRACE_AVAILABLE</code> , aber nicht die Funktion <code>AHAFS_CALLBACK_INTRCNTX</code> angegeben ist. |

Tabelle 74. Schlüsselwörter (Forts.)

| Schlüssel | Wert | Details |
|--------------------------------------|--|---|
| UID_LOGIN | Integer | Die Anmelde-Benutzer-ID des Benutzers, der das Ereignisvorkommen ausgelöst hat. Die ID ist nur bei einem Ereigniserzeuger verfügbar, für den die Funktion AHAFS_STKTRACE_AVAILABLE , aber nicht die Funktion AHAFS_CALLBACK_INTRCNTX angegeben ist. |
| GID | Integer | Die aktuelle Gruppen-ID des Benutzers, der das Ereignisvorkommen ausgelöst hat. Die ID ist nur bei einem Ereigniserzeuger verfügbar, für den die Funktion AHAFS_STKTRACE_AVAILABLE , aber nicht die Funktion AHAFS_CALLBACK_INTRCNTX angegeben ist. |
| PROG_NAME | String | Der Name des Prozesses, der das Ereignisvorkommen ausgelöst hat. Die ID ist nur bei einem Ereigniserzeuger verfügbar, für den die Funktion AHAFS_STKTRACE_AVAILABLE , aber nicht die Funktion AHAFS_CALLBACK_INTRCNTX angegeben ist. |
| CURRENT_VALUE | 64-Bit-Ganzzahl ohne Vorzeichen im Dezimalformat | Dieser Schlüssel ist nur für AHAFS_THRESHOLD_VALUE -Ereigniserzeuger bestimmt und gibt den aktuellen Wert des Ereignisses zu dem Zeitpunkt, zu dem das Ereignisvorkommen erkannt wurde, zurück. Aufgrund der Verzögerung zwischen der Benachrichtigung eines Prozesses und dem Lesen der Ereignisdaten kann sich der tatsächliche aktuelle Wert des Ereignisses geändert haben. |
| RC_FROM_EVPROD | 32-Bit-Ganzzahl im Dezimalformat | Dieser Rückkehrcode wird vom Ereigniserzeuger als Ergebnis eines Fehlers beim Versuch, das Ereignis zu konfigurieren, oder als Ergebnis eines Ereignisvorkommens zurückgegeben. Im Allgemeinen zeigen Rückkehrcodes kleiner als 256 einen Fehler an, wenn versucht wird, das Ereignis beim Ereigniserzeuger zu registrieren. Einige Ereigniserzeuger geben Codes größer als 256 zurück, um mehr Informationen zum Ereignisvorkommen bereitzustellen. Diese Rückkehrcodes sind in sys/ahafs_evProds.h dokumentiert. |
| BEGIN_EVPROD_INFO
END_EVPROD_INFO | Zeichenfolge* | Diese beiden Schlüsselwörter markieren den Anfang und das Ende der Zeichenfolge, die vom Ereigniserzeuger zurückgegeben wird. Hinter BEGIN_EVPROD_INFO und vor END_EVPROD_INFO wird immer ein Zeilenumbruch eingefügt. Für Konsumenten, für die CLUSTER=YES ist, werden Knoteninformationen bereitgestellt. |

Tabelle 74. Schlüsselwörter (Forts.)

| Schlüssel | Wert | Details |
|---|---------------|---|
| STACK_TRACE | Zeichenfolge* | Für Konsumenten, die INFO_LVL=3 mit einem Ereigniserzeuger angeben, für den die Funktion AHAFS_STKTRACE_AVAILABLE , aber nicht die Funktion AHAFS_CALLBACK_INTRCNTX angegeben ist, wird der Stack-Trace des Ereignisvorkommens bereitgestellt. Das Schlüsselwort STACK_TRACE zeigt an, dass die verbleibenden Ereignisdaten bis zur Zeichenfolge END_EVENT_INFO den Stack des Ereignisvorkommens darstellen. |
| NUM_EVDROPS_INTRCNTX | Integer | Dieses Schlüsselwort stellt die Anzahl verworfener Vorkommen von Interruptkontextereignissen seit dem Zeitpunkt dar, der in den Feldern TIME0_tvsec und TIME0_tvnsec angegeben wurde. Die Ereignisvorkommen werden nur verworfen, wenn die Anzahl der Ereignisvorkommen sehr hoch ist. |
| TIME0_tvsec
TIME0_tvnsec | Integer | Diese Schlüsselwörter zeichnen die Zeitmarke des ersten verworfenen Ereignisvorkommens und die Nanosekunden seit der Epoche auf. Diese Schlüsselwörter werden zusammen mit dem Schlüsselwort NUM_EVDROPS_INTRCNTX berichtet. |

Konsolidierung doppelter Ereignisse:

Wenn dasselbe Ereignis mehrfach eintritt, bevor der Konsument die Daten gelesen hat, werden die doppelten Einträge zu einem einzigen Eintrag konsolidiert. Diese Konsolidierung wird durch nicht sequenzielle Folge Nummern ohne entsprechendes **BUF_WRAP**-Schlüsselwort angezeigt. Die Zeitmarke und die Folge Nummern spiegeln das letzte Vorkommen des Ereignisses wider.

Beispielereignisdaten

Für einen Ereigniserzeuger, der **AHAFS_THRESHOLD_STATE** und **AHAFS_STKTRACE_AVAILABLE** angegeben hat und eine Nachricht an Ereigniskonsumenten übergibt, sehen die drei Ebenen der Ausgabe wie folgt aus:

| INFO_LVL=1 | INFO_LVL=2 | INFO_LVL=3 |
|---|---|---|
| <pre> BEGIN_EVENT_INFO TIME_tvsec=1269863383 TIME_tvnsec=455993143 SEQUENCE_NUM=0 PID=6947038 UID=0 UID_LOGIN=0 GID=0 PROG_NAME=cat RC_FROM_EVPROD=1000 END_EVENT_INFO </pre> | <pre> BEGIN_EVENT_INFO TIME_tvsec=1269863383 TIME_tvnsec=455993143 SEQUENCE_NUM=0 PID=6947038 UID=0 UID_LOGIN=0 GID=0 PROG_NAME=cat RC_FROM_EVPROD=1000 BEGIN_EVPROD_INFO event producer message here END_EVPROD_INFO END_EVENT_INFO </pre> | <pre> BEGIN_EVENT_INFO TIME_tvsec=1269863383 TIME_tvnsec=455993143 SEQUENCE_NUM=0 PID=6947038 UID=0 UID_LOGIN=0 GID=0 PROG_NAME=cat RC_FROM_EVPROD=1000 BEGIN_EVPROD_INFO event producer message here END_EVPROD_INFO STACK_TRACE ahafs_prod_callback+3C4 ahafs_cbfm_wrapper+30 ahafs_vn_write+204 vno_rdw+7E4 vno_rw+B4 rwuio+12C rdwr+184 kewrite+16C .svc_instr write+1A4 _xwrite+6C _xflsbuf+B0 __flsbuf+9C copyopt_ascii+2C0 scat+388 main+11C _start+68 END_EVENT_INFO </pre> |

Für einen Ereigniserzeuger, der **AHAFS_THRESHOLD_VALUE_HI** angegeben hat, aber **AHAFS_STKT-RACE_AVAILABLE** nicht und eine Nachricht an Ereigniskonsumenten übergibt, sehen die drei Ebenen der Ausgabe wie folgt aus:

| INFO_LVL=1 | INFO_LVL=2 | INFO_LVL=3 |
|--|--|--|
| <pre> BEGIN_EVENT_INFO TIME_tvsec=1269866715 TIME_tvnsec=16678418 SEQUENCE_NUM=0 CURRENT_VALUE=3 RC_FROM_EVPROD=1000 END_EVENT_INFO </pre> | <pre> BEGIN_EVENT_INFO TIME_tvsec=1269866715 TIME_tvnsec=16678418 SEQUENCE_NUM=0 CURRENT_VALUE=3 RC_FROM_EVPROD=1000 BEGIN_EVPROD_INFO event producer message here END_EVPROD_INFO END_EVENT_INFO </pre> | <pre> BEGIN_EVENT_INFO TIME_tvsec=1269866715 TIME_tvnsec=16678418 SEQUENCE_NUM=0 CURRENT_VALUE=3 RC_FROM_EVPROD=1000 BEGIN_EVPROD_INFO event producer message here END_EVPROD_INFO END_EVENT_INFO </pre> |

Fehlerformat:

Wenn ein Fehler vom Ereigniserzeuger ausgegeben wird, haben alle Ereigniserzeuger das folgende Format für alle **INFO_LVL**-Ereignisse:

```

BEGIN_EVENT_INFO
TIME_tvsec=1269868036
TIME_tvnsec=966708948
SEQUENCE_NUM=0
RC_FROM_EVPROD=20
END_EVENT_INFO

```

Wenn ein Konsument ein **AHAFS_THRESHOLD_VALUE**-Ereignis überwacht und der aktuelle Wert den angeforderten Schwellenwert bereits überschreitet, wird das Fehlerformat auch für die Aufzeichnung dieses **EALREADY**-Ereignisses verwendet:

```

BEGIN_EVENT_INFO
TIME_tvsec=1269868036
TIME_tvnsec=966708948
SEQUENCE_NUM=0
CURRENT_VALUE=1
RC_FROM_EVPROD=56
END_EVENT_INFO

```

BUF_WRAP und EVENT_OVERFLOW:

Wenn ungelesene Daten mit Daten eines neuen Vorkommens des Ereignisses überschrieben werden, ist das Schlüsselwort **BUF_WRAP** die erste Ausgabe eines Aufrufs **read()** für die Monitordatei. Gibt es einen Pufferumlauf und einen Ereignisüberlauf, wird immer zuerst **BUF_WRAP** und dann erst **EVENT_OVERFLOW** ausgegeben. Im Folgenden sehen Sie eine Beispielausgabe von **read()** für den Fall eines Pufferumlaufs und eines Ereignisüberlaufs:

Zuerst gibt **read()** Folgendes zurück:

```
BUF_WRAP
```

Anschließend gibt **read()** Folgendes zurück:

```
EVENT_OVERFLOW
```

Danach gibt **read()** die Ereignisdaten zurück, die in den Puffer gepasst haben:

```

BEGIN_EVENT_INFO
TIME_tvsec=1269863383
TIME_tvnsec=455993143
SEQUENCE_NUM=0
PID=6947038
UID=0
UID_LOGIN=0
GID=0
PROG_NAME=cat
RC_FROM_EVPROD=1000
BEGIN_EVPROD_INFO
event producer message here
END_EVPROD_INFO
STACK_TRACE
ahafs_prod_callback+3C4
ahafs_cbfm_wrapper+30
ahafs_vn_write+204
vno_rdw+7E4
vno_rw+B4
rwuio+12C
rdwr+184
kewrite+16C
.svc_instr
write+1A4
_xwri

```

Treffen die Ereignisdaten schnell genug ein, ist es möglich, dass zwei **BUF_WRAP**-Einträge nacheinander empfangen werden. Wenn Sie **BUF_WRAP** sehen, erhöhen Sie die Puffergröße (mit **BUF_SIZE** beim Schreiben in die Monitordatei).

NUM_EVDROPS_INTRCNTX:

Falls ein Vorkommen eines Interruptkontextereignisses aufgrund einer hohen Anzahl an Ereignisvorkommen verworfen wird, enthält die Ausgabe eines Aufrufs der Methode **read()** für die Ereignisdatei, die dieses Ereignis darstellt, das Schlüsselwort **NUM_EVDROPS_INTRCNTX** unmittelbar hinter der Zeile, die das Schlüsselwort **BEGIN_EVENT_INFO** enthält.

Das folgende Beispiel stellt die Ausgabe eines Aufrufs der Methode **read()** dar:

```

BEGIN_EVENT_INFO
BEGIN_EVENT_INFO
NUM_EVDROPS_INTRCNTX=5508
TIME0_tvsec=1353437661
TIME0_tvnsec=75494625
TIME_tvsec=1353437661
TIME_tvnsec=741365037
SEQUENCE_NUM=6663
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
...msg from event-producer...
END_EVPROD_INFO
END_EVENT_INFO

```

Diese Beispielausgabe enthält die folgenden Gruppen von Informationen:

- Der Wert `NUM_EVDROPS_INTRCNTX=5508` ist die Anzahl verworfener Vorkommen von Interruptkontextereignissen seit dem Zeitpunkt, der in den Feldern `TIME0_tvsec` und `TIME0_tvnsec` angegeben wurde.
- Die verbleibenden Informationen (d. h. `SEQUENCE_NUM=6663`, `RC_FROM_EVPROD=0`, `...msg from event-producer...` usw.) beziehen sich auf das Ereignis, das zu dem Zeitpunkt eingetreten ist, der in den Feldern `TIME_tvsec` und `TIME_tvnsec` angegeben wurde.

Vordefinierte Ereigniserzeuger

modFile

Der Ereigniserzeuger `modFile` überwacht Änderungen von Dateiinhalten.

Übersicht

Der Ereigniserzeuger `modFile` befindet sich im Verzeichnis `fs` und überwacht Änderungen von Dateiinhalten. Die folgenden `vnode`-Operationen werden überwacht: `vnop_rdwrt()`, `vnop_map_lloff()`, `vnop_remove()`, `vnop_ftrunc()`, `vnop_fclear()` und `vnop_rename()`. Änderungen, die nicht über die LFS-Ebene vorgenommen werden, können nicht überwacht werden (d. h. Schreiboperationen in zugeordnete Dateien).

Dateien können nicht überwacht werden, wenn

- sie sich in einem fernen Dateisystem befinden.
- sie sich in einem Dateisystem des Typs `ahafs`, `procfs` oder `namefs` befinden.
- sie symbolische Verbindungen sind.
- sie sich in einem Verzeichnis befinden, das mit einer Erweiterung von AIX Event Infrastructure endet (`.mon`, `.list`, `.monFactory`).
- Monitordateien mit einem vollständigen Pfadnamen, der länger ist als der mit `MAXPATHLEN` im Pseudodateisystem von AIX Event Infrastructure festgelegte Wert, können nicht überwacht werden.

Funktionen

```

AHAFS_THRESHOLD_STATE
AHAFS_STKTRACE_AVAILABLE
AHAFS_REMOTE_EVENT_ENABLED

```

Rückkehrcodes

Der Ereigniserzeuger `modFile` verwendet Rückkehrcodes, die in `<sys/ahafs_evProds.h>` definiert sind.

Diese Rückkehrcodes werden verwendet, um anzuzeigen, wie der Inhalt des überwachten Verzeichnisses geändert wurde:

AHAFS_MODFILE_WRITE

Es wurde in die überwachte Datei geschrieben.

AHAFS_MODFILE_UNMOUNT

Das Dateisystem mit der überwachten Datei wurde abgehängt. Dies ist ein Nichtverfügbarkeitsereignis.

AHAFS_MODFILE_MAP

Ein Prozess hat einen Teil der überwachten Datei zum Schreiben zugeordnet.

AHAFS_MODFILE_REMOVE

Die überwachte Datei wurde entfernt. Dies ist ein Nichtverfügbarkeitsereignis.

AHAFS_MODFILE_RENAME

Die überwachte Datei wurde umbenannt. Dies ist ein Nichtverfügbarkeitsereignis.

AHAFS_MODFILE_FCLEAR

Ein Prozess hat ein **fclear** für die überwachte Datei abgesetzt.

AHAFS_MODFILE_FTRUNC

Ein Prozess hat ein **ftrunc** für die überwachte Datei abgesetzt.

AHAFS_MODFILE_OVERMOUNT

Die überwachte Datei wurde zu oft angehängt.

Nachrichten des Ereigniserzeugers

Diese Ereigniserzeuger übergibt keine Nachrichten im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Für die Überwachung von Dateiänderungen muss eine Monitordatei im Verzeichnis **modFile.monFactory** erstellt werden, die denselben Pfad hat wie die zu überwachende Datei. Wenn Sie beispielsweise **/etc/passwd** überwachen möchten, wird die Datei **/aha/fs/modFile.monFactory/etc/passwd.mon** verwendet.

Beispielereignisdaten

Die folgenden Ereignisdaten wurden über einen Prozess generiert, der in eine überwachte Datei geschrieben hat. Dies ist die Ausgabe mit der **INFO_LVL**-Stufe 3:

```
BEGIN_EVENT_INFO
TIME_tvsec=1271703118
TIME_tvnsec=409201093
SEQUENCE_NUM=0
PID=5701678
UID=0
UID_LOGIN=0
GID=0
PROG_NAME=cat
RC_FROM_EVPROD=1000
STACK_TRACE
aha_cbfn_wrapper+30
ahafs_evprods+510
aha_vn_write+154
vno_p_rdr+7E8
vno_rw+B4
rwuio+100
rdwr+188
kewrite+104
.svc_instr
write+1A4
_xwrite+6C
_xflsbuf+A8
__flsbuf+C0
copyopt+2E8
scat+22C
main+11C
_start+68
END_EVENT_INFO
```

modFileAttr

Der Ereigniserzeuger **modFileAttr** überwacht Änderungen, die an den Attributen einer Datei vorgenommen werden.

Übersicht

Der Ereigniserzeuger **modFileAttr** befindet sich im Verzeichnis **fs** und überwacht Änderungen, die an den Attributen einer Datei oder eines Verzeichnisses vorgenommen werden (Modus, Eigner und Zugriffssteuerungslisten). Die folgenden vnode-Operationen werden überwacht: **vnop_setattr()** (nur für die Operationen **V_OWN** und **V_MODE**) **vnop_setacl()**, **vnop_setxacl()**, **vnop_remove()**, **vnop_rename()** und **vnop_rmdir()**.

Dateien und Verzeichnisse können nicht überwacht werden, wenn

- sie sich in einem fernen Dateisystem befinden,
- sie sich in einem Dateisystem des Typs **ahafs**, **procfs** oder **namefs** befinden,
- sie sich in einem Verzeichnis befinden, das mit einer Erweiterung von AIX Event Infrastructure endet (**.mon**, **.list**, **.monFactory**),
- Monitordateien mit einem vollständigen Pfadnamen, der länger ist als der mit **MAXPATHLEN** im Pseudodateisystem von AIX Event Infrastructure festgelegte Wert, können nicht überwacht werden.

Funktionen

AHAFS_THRESHOLD_STATE

AHAFS_STKTRACE_AVAILABLE

AHAFS_REMOTE_EVENT_ENABLED

Rückkehrcodes

Der Ereigniserzeuger **modFileAttr** verwendet Rückkehrcodes, die in `<sys/ahafs_evProds.h>` definiert sind.

Diese Rückkehrcodes werden verwendet, um anzuzeigen, wie der Inhalt des überwachten Verzeichnisses geändert wurde:

AHAFS_MODFILEATTR_UNMOUNT

Das Dateisystem mit der überwachten Datei bzw. dem überwachten Verzeichnis wurde abgehängt. Dies ist ein Nichtverfügbarkeitsereignis.

AHAFS_MODFILEATTR_REMOVE

Die überwachte Datei bzw. das überwachte Verzeichnis wurde entfernt. Dies ist ein Nichtverfügbarkeitsereignis.

AHAFS_MODFILEATTR_RENAME

Die überwachte Datei bzw. das überwachte Verzeichnis wurde umbenannt. Dies ist ein Nichtverfügbarkeitsereignis.

AHAFS_MODFILEATTR_OVERMOUNT

Die überwachte Datei bzw. das überwachte Verzeichnis wurde zu oft angehängt. Dies ist ein Nichtverfügbarkeitsereignis.

AHAFS_MODFILEATTR_SETACL

Die Zugriffssteuerungslisten der überwachten Datei bzw. des überwachten Verzeichnisses wurden geändert.

AHAFS_MODFILEATTR_SETOWN

Der Eigner der überwachten Datei bzw. des überwachten Verzeichnisses wurde geändert.

AHAFS_MODFILEATTR_SETMODE

Der Modus der überwachten Datei bzw. des überwachten Verzeichnisses wurde geändert.

Nachrichten des Ereigniserzeugers

Diese Ereigniserzeuger übergibt keine Nachrichten im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Für die Überwachung von Dateiänderungen muss eine Monitordatei im Verzeichnis **modFileAttr.monFactory** erstellt werden, die denselben Pfad hat wie die zu überwachende Datei. Wenn Sie beispielsweise **/etc/passwd** überwachen möchten, wird die Datei **/aha/fs/modFileAttr.monFactory/etc/passwd.mon** verwendet.

Beispielereignisdaten

Die folgenden Ereignisdaten wurden von einem Prozess generiert, der den Modus einer überwachten Datei ändert. Dies ist die Ausgabe mit der **INFO_LVL**-Stufe 3:

```
BEGIN_EVENT_INFO
TIME_tvsec=1291994430
TIME_tvnsec=760097298
SEQUENCE_NUM=0
PID=5767216
UID=0
UID_LOGIN=0
GID=0
PROG_NAME=chmod
RC_FROM_EVPROD=1010
STACK_TRACE
ahafs_evprods+70C
aha_process_attr+120
vnode_setattr+21C
vsetattr@AF13_1+20
setnameattr+B4
chmod+110
.svc_instr
change+3C8
main+190
_start+68
END_EVENT_INFO
```

modDir

Der Ereigniserzeuger **modDir** überwacht Änderungen von Verzeichnisinhalten.

Übersicht

Der Ereigniserzeuger **modDir** befindet sich im Verzeichnis **fs** und überwacht Änderungen von Verzeichnisinhalten. Die folgenden **vnode**-Operationen werden überwacht: **vnode_create()**, **vnode_link()**, **vnode_symlink()**, **vnode_remove()**, **vnode_rename()**, **vnode_mkdir()** und **vnode_rmdir()**.

Verzeichnisse können nicht überwacht werden, wenn

- sie sich in einem fernen Dateisystem befinden.
- sie sich in einem Dateisystem des Typs **ahafs**, **procfs** oder **namefs** befinden.
- sie symbolische Verbindungen sind.
- sie sich in einem Verzeichnis befinden, das mit einer Erweiterung von AIX Event Infrastructure endet (**.mon**, **.list**, **.monFactory**).
- Monitordateien mit einem vollständigen Pfadnamen, der länger ist als der mit **MAXPATHLEN** im Pseudodateisystem von AIX Event Infrastructure festgelegte Wert, können nicht überwacht werden.

Der Ereigniserzeuger **modDir** führt keine rekursive Überwachung von Verzeichnisänderungen durch. Es werden nur Änderungen am angegebenen Verzeichnis überwacht.

Funktionen

```
AHAFS_THRESHOLD_STATE
AHAFS_STKTRACE_AVAILABLE
AHAFS_REMOTE_EVENT_ENABLED
```

Rückkehrcodes

Der Ereigniserzeuger **modDir** verwendet Rückkehrcodes, die in `<sys/ahafs_evProds.h>` definiert sind.

Diese Rückkehrcodes werden verwendet, um anzuzeigen, wie der Inhalt des überwachten Verzeichnisses geändert wurde:

AHAFS_MODDIR_CREATE

Ein neues Dateisystemobjekt wurde unter dem überwachten Verzeichnis erstellt.

AHAFS_MODDIR_UNMOUNT

Das Dateisystem mit dem überwachten Verzeichnis wurde abgehängt. Dies ist ein Nichtverfügbarkeitsereignis.

AHAFS_MODDIR_REMOVE

Ein Dateisystemobjekt im überwachten Verzeichnis wurde entfernt.

AHAFS_MODDIR_REMOVE_SELF

Das überwachte Verzeichnis selbst wurde entfernt oder umbenannt. Dies ist ein Nichtverfügbarkeitsereignis.

Nachrichten des Ereigniserzeugers

Der Name des Dateisystemobjekts, das das Ereignis ausgelöst hat, wird in die Ereignisdaten eingeschlossen.

Zulässige Monitordateien

Für die Überwachung von Änderungen von Verzeichnisinhalten muss eine Monitordatei im Verzeichnis **modDir.monFactory** erstellt werden, die denselben Pfad wie das zu überwachende Verzeichnis hat. Wenn Sie beispielsweise das Verzeichnis `/home/cheryl` auf Änderungen überwachen möchten, wird die Monitordatei `/aha/fs/modDir.monFactory/home/cheryl.mon` verwendet.

Beispielereignisdaten

Die folgenden Ereignisdaten wurden aus einer neuen Datei mit dem Namen **file1** generiert, die in einem überwachten Verzeichnis erstellt wurde. Dies ist die Ausgabe mit der **INFO_LVL**-Stufe 3:

```
BEGIN_EVENT_INFO
TIME_tvsec=1271780397
TIME_tvnssec=24369022
SEQUENCE_NUM=0
PID=6095102
UID=0
UID_LOGIN=0
GID=0
PROG_NAME=touch
RC_FROM_EVPROD=1000
BEGIN_EVPROD_INFO
file1
END_EVPROD_INFO
STACK_TRACE
aha_cbfm_wrapper+30
ahafs_evprods+510
aha_process_vnop+138
vnop_create_attr+4AC
openpnp+418
openpath+100
copen+294
kopen+1C
.svc_instr
open+F8
creat64+1C
main+1EC
_start+68
END_EVENT_INFO
```

utilFs

Der Ereigniserzeuger **utilFs** überwacht die Belegung eines Dateisystems.

Übersicht

Der Ereigniserzeuger **utilFs** überwacht die Belegung eines Dateisystems in Prozent. Er befindet sich im Verzeichnis **fs**. Derzeit unterstützen nur JFS2-Dateisysteme die **utilFs**-Überwachung. Bei jedem Schreiben in eine Datei, Erstellen einer Datei oder Löschen einer Datei wird die Belegung des Dateisystems geprüft, um festzustellen, ob der angegebene Schwellenwert erreicht oder überschritten wurde. Einige dateisystemspezifische Operationen können sich auf die Belegung des Dateisystems auswirken, aber **utilFs** kann diese unter Umständen erst beim nächsten Schreiben in eine Datei, Erstellen einer Datei oder Löschen einer Datei erkennen. Schwellenwerte, die durch das Löschen eines Dateiobjekts überschritten, werden erst beim nächsten Schreiben einer Datei, Erstellen einer Datei oder Löschen einer Datei gemeldet.

Dateisysteme, die eine Monitordatei mit einem Pfadnamen haben, der länger ist als der mit **MAXPATHLEN** im AHAFS festgelegte Wert, können nicht überwacht werden.

Um eine Flut von Ereignisbenachrichtigungen und potenziellen Leistungsproblemen zu vermeiden, wird dringend empfohlen, **utilFs**-Ereignisse mit der **NOTIFY_CNT**-Einstellung 1 zu überwachen.

Funktionen

```
AHAFS_THRESHOLD_VALUE_HIGH
AHAFS_THRESHOLD_VALUE_LOW
AHAFS_REMOTE_EVENT_ENABLED
```

Angegebene Schwellenwerte müssen zwischen 1 und 100 inklusive liegen.

Rückkehrcodes

Der Ereigniserzeuger **utilFs** verwendet Rückkehrcodes, die in `<sys/ahafs_evProds.h>` definiert sind.

Diese Rückkehrcodes werden verwendet, um anzuzeigen, wie der Inhalt des überwachten Verzeichnisses geändert wurde:

AHAFS_UTILFS_THRESH_HIT

Das überwachte Dateisystem hat den angegebenen Schwellenwert erreicht.

AHAFS_UTILFS_UNMOUNT

Das überwachte Dateisystem wurde abgehängt. Dies ist ein Nichtverfügbarkeitsereignis.

Nachrichten des Ereigniserzeugers

Diese Ereigniserzeuger übergibt keine Nachrichten im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Für die Überwachung der Dateisystembelegung muss eine Monitordatei im Verzeichnis **utilFs**-**.monFactory** erstellt werden, die denselben Pfad hat wie der Mountpunkt des zu überwachenden Dateisystems. Wenn Sie beispielsweise das Dateisystem **/data/fs1** überwachen möchten, wird die Datei **/aha/fs/utilFs.monFactory/data/fs1.mon** verwendet.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **AHAFS_UTILFS_THRESH_HIT**-Ereignisses für die **INFO_LVL**-Einstellung 3:

```
BEGIN_EVENT_INFO
TIME_tvsec=1271705858
TIME_tvnsec=704241888
SEQUENCE_NUM=0
CURRENT_VALUE=10
RC_FROM_EVPROD=1000
END_EVENT_INFO
```

waitTmCPU

Der Ereigniserzeuger **waitTmCPU** überwacht die durchschnittliche Wartezeit ausführbarer Threads.

Übersicht

Der Ereigniserzeuger **waitTmCPU** überwacht die durchschnittliche Wartezeit (Millisekunden) ausführbarer Threads, die auf CPU-Zeit warten, im Sekundenintervall. **waitTmCPU** befindet sich im Verzeichnis **cpu**.

Funktionen

```
AHAFS_THRESHOLD_VALUE_HIGH
AHAFS_CALLBACK_INTRCNTX
AHAFS_REMOTE_EVENT_ENABLED
```

Angegebene Schwellenwerte müssen größer sein als 0.

Rückkehrcodes

Dieser Ereigniserzeuger gibt immer 0 zurück, wenn das Ereignis eintritt.

Nachrichten des Ereigniserzeugers

Diese Ereigniserzeuger übergibt keine Nachrichten im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Für die Überwachung dieses Ereignisses muss die folgende Monitordatei verwendet werden:
/aha/cpu/waitTmCPU.monFactory/waitTmCPU.mon

In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **waitTmCPU**-Ereignisses mit der **INFO_LVL**-Einstellung 3:

```
BEGIN_EVENT_INFO
TIME_tvsec=1271779504
TIME_tvnsec=18056777
SEQUENCE_NUM=0
CURRENT_VALUE=4
RC_FROM_EVPROD=0
END_EVENT_INFO
```

waitersFreePg

Der Ereigniserzeuger **waitersFreePg** überwacht die Anzahl der Threads, die auf einen freien Rahmen warten.

Übersicht

Der Ereigniserzeuger **waitersFreePg** überwacht die Anzahl der Threads, die auf einen freien Rahmen warten, im Sekundenintervall. **waitersFreePg** befindet sich im Unterverzeichnis **mem**.

Funktionen

```
AHAFS_THRESHOLD_VALUE_HIGH
AHAFS_CALLBACK_INTRCNTX
AHAFS_REMOTE_EVENT_ENABLED
```

Angegebene Schwellenwerte müssen größer sein als 0.

Rückkehrcodes

Dieser Ereigniserzeuger gibt immer 0 zurück, wenn das Ereignis eintritt.

Nachrichten des Ereigniserzeugers

Diese Ereigniserzeuger übergibt keine Nachrichten im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Für die Überwachung dieses Ereignisses muss die folgende Monitordatei verwendet werden:

/aha/mem/waitersFreePg.monFactory/waitersFreePg.mon

In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **waitersFreePg**-Ereignisses mit der **INFO_LVL**-Einstellung 3:

```
BEGIN_EVENT_INFO
TIME_tvsec=1271779680
TIME_tvnsec=347233732
SEQUENCE_NUM=0
CURRENT_VALUE=19843
RC_FROM_EVPROD=0
END_EVENT_INFO
```

waitTmPgInOut

Der Ereigniserzeuger **waitTmPgInOut** überwacht die durchschnittliche Wartezeit in Millisekunden für Threads, die auf eine Seiteneinlagerungs- bzw. Seitenauslagerungsoperation warten.

Übersicht

Der Ereigniserzeuger **waitTmPgInOut** überwacht die durchschnittliche Wartezeit (Millisekunden) für Threads, die auf den Abschluss einer Seiteneinlagerungs- bzw. Seitenauslagerungsoperation warten, über einen Zeitraum von einer Sekunde. Der Ereigniserzeuger **waitTmPgInOut** befindet sich im Verzeichnis **mem**.

Funktionen

```
AHAFS_THRESHOLD_VALUE_HIGH
AHAFS_CALLBACK_INTRCNTX
AHAFS_REMOTE_EVENT_ENABLED
```

Angegebene Schwellenwerte müssen größer sein als 0.

Rückkehrcodes

Dieser Ereigniserzeuger gibt immer 0 zurück, wenn das Ereignis eintritt.

Nachrichten des Ereigniserzeugers

Diese Ereigniserzeuger übergibt keine Nachrichten im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Für die Überwachung dieses Ereignisses muss die folgende Monitordatei verwendet werden:

/aha/mem/waitTmPgInOut.monFactory/waitTmPgInOut.mon

In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **waitTmPgInOut**-Ereignisses mit der **INFO_LVL**-Einstellung 3:

```
BEGIN_EVENT_INFO
TIME_tvsec=1271779359
TIME_tvnsec=941699413
SEQUENCE_NUM=0
CURRENT_VALUE=12
RC_FROM_EVPROD=0
END_EVENT_INFO
```

vmo

Der Ereigniserzeuger **vmo** überwacht Änderungen an den optimierbaren **vmo**-Parametern.

Übersicht

Der Ereigniserzeuger **vmo** befindet sich im Verzeichnis **mem** und überwacht Änderungen an den folgenden optimierbaren **vmo**-Parametern.

Anmerkung: Der Befehl **vmo** ist ein selbstdokumentierender Befehl. Einige der im Folgenden aufgelisteten optimierbaren Parameter werden möglicherweise nicht unterstützt.

- **npskill**
- **npswarn**
- **force_relalias_lite**
- **low_ps_handling**
- **maxpin%** (muss mit der Datei **maxpin_pct.mon** überwacht werden)
- **nokilluid**
- **relalias_percentage**
- **vmm_default_pspa**
- **npsrpgmin**
- **npsrpgmax**
- **npsscrubmin**
- **npsscrubmax**
- **scrubclean**
- **rpgcontrol**
- **rpgclean**
- **vm_modlist_threshold**
- **vmm_fork_policy**
- **lru_poll_interval**

Funktionen

```
AHAFS_THRESHOLD_STATE
AHAFS_STKTRACE_AVAILABLE
AHAFS_REMOTE_EVENT_ENABLED
```

Rückkehrcodes

Dieser Ereigniserzeuger gibt immer 0 zurück, wenn das Ereignis eintritt.

Nachrichten des Ereigniserzeugers

Diese Ereigniserzeuger übergibt keine Nachrichten im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Für die Überwachung der zuvor genannten optimierbaren Parameter müssen Monitordateien im folgenden Format verwendet werden:

```
/aha/mem/vmo.monFactory/<optimierbarer_Parameter>.mon
```

Dateien, die den zuvor genannten Ereignissen nicht entsprechen, können in diesem Verzeichnis nicht erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten einer Änderung eines überwachten optimierbaren Parameters mit der **INFO_LVL**-Einstellung 3.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271770698
TIME_tvnsec=787565808
SEQUENCE_NUM=0
PID=5701808
UID=0
UID_LOGIN=0
GID=0
```

```

PROG_NAME=vmo
RC_FROM_EVPROD=0
STACK_TRACE
aha_cbfm_wrapper+30
ahafs_evprods+510
vm_mon_tunable+B0
vm_chk_mod_tun+5CC
_vmgetinfo+53C
vmgetinfo+48
.svc_instr
vmo_write_vmsetkervars+134
vmo_write_dynamic_values+404
main+BC
__start+70
END_EVENT_INFO

```

schedo

Dieser Ereigniserzeuger überwacht Änderungen an optimierbaren **schedo**-Parametern.

Übersicht

Derzeit kann nur der optimierbare Parameter **vpm_xvcpus** überwacht werden. Dieser Ereigniserzeuger gibt einen Stack-Trace und Benutzerinformationen zurück, wenn das Ereignis eintritt. Dieser Ereigniserzeuger befindet sich im Verzeichnis **cpu**.

Funktionen

```

AHAFS_THRESHOLD_STATE
AHAFS_STKTRACE_AVAILABLE
AHAFS_REMOTE_EVENT_ENABLED

```

Rückkehrcodes

Dieser Ereigniserzeuger gibt immer 0 zurück, wenn das Ereignis eintritt.

Nachrichten des Ereigniserzeugers

Diese Ereigniserzeuger übergibt keine Nachrichten im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Die für die Überwachung dieses optimierbaren Parameters verwendete Monitordatei ist:

```
/aha/cpu/schedo.monFactory/vpm_xvcpus.mon
```

In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten zur Änderung des optimierbaren Parameters **vpm_xvcpus** mit der **INFO_LVL**-Einstellung 3:

```

BEGIN_EVENT_INFO
TIME_tvsec=1271771009
TIME_tvnsec=251723285
SEQUENCE_NUM=0
PID=7143474
UID=0
UID_LOGIN=0
GID=0
PROG_NAME=schedo
RC_FROM_EVPROD=0
STACK_TRACE
aha_cbfm_wrapper+30
ahafs_evprods+510
schedtune+394
.svc_instr
schedo_write_schedparams+94
schedo_write_dynamic_values+6F0
main+1B0
__start+68
END_EVENT_INFO

```

pidProcessMon

Der Ereigniserzeuger **pidProcessMon** überwacht die Beendigung von Prozessen basierend auf der PID.

Übersicht

Der Ereigniserzeuger **pidProcessMon** befindet sich im Verzeichnis **cpu** und überwacht die Beendigung von Prozessen basierend auf der PID.

Funktionen

AHAFS_THRESHOLD_STATE
AHAFS_CALLBACK_INTRCNTX

Rückkehrcodes

Der Ereigniserzeuger **pidProcessMon** gibt nur den Rückkehrcode 0 zurück.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachricht **PROCESS_DOWN** im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Für die Überwachung von Prozessbeendigungen muss eine Monitordatei im Verzeichnis **pidProcessMon.monFactory** erstellt werden. Der Name der Monitordatei muss im Format

```
/aha/cpu/pidProcessMon.monFactory/<Prozess-PID>.mon
```

angegeben werden.

Beispielereignisdaten

Die folgenden Ereignisdaten wurden bei der Beendigung eines überwachten Prozesses generiert. Dies ist die Ausgabe mit der Einstellung **INFO_LVL**.

```
BEGIN_EVENT_INFO  
TIME_tvsec=1272348759  
TIME_tvnsec=379259175  
SEQUENCE_NUM=0  
RC_FROM_EVPROD=0  
BEGIN_EVPROD_INFO  
EVENT_TYPE=PROCESS_DOWN  
END_EVPROD_INFO  
END_EVENT_INFO
```

processMon

Der Ereigniserzeuger **processMon** überwacht die Beendigung von Prozessen.

Übersicht

Der Ereigniserzeuger **processMon** befindet sich im Verzeichnis "cpu" und überwacht die Beendigung von Prozessen basierend auf dem Prozessnamen. Es wird nur der übergeordnete Prozess für einen bestimmten Prozess mit demselben Namen überwacht. Wenn Sie beispielsweise eine Prozessstruktur **abc (pid 123)->xyz (pid 345)->xyz (pid 567)** haben und jemand die Überwachung des Prozesses **xyz** anfordert, wird (**pid = 345**) überwacht.

Funktionen

AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX

Rückkehrcodes

Der Ereigniserzeuger **processMon** gibt nur den Rückkehrcode 0 zurück.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachricht **PROCESS_DOWN** im Rahmen seiner Ereignisdaten.

Zulässige Monitordateien

Für die Überwachung von Prozessbeendigungen muss eine Monitordatei im Verzeichnis **process-**

Mon.monFactory erstellt werden, die denselben Pfad hat wie die Datei, die zum Starten des Prozesses verwendet wurde. Wenn Sie beispielsweise einen Prozess mit dem Namen **test** im Verzeichnis **/usr/samples/ahafs** überwachen möchten, wird die Monitordatei **/aha/cpu/processMon.monFactory/usr/samples/ahafs/test.mon** verwendet.

Beispielereignisdaten

Die folgenden Ereignisdaten wurden bei der Beendigung eines überwachten Prozesses generiert. Dies ist die Ausgabe mit der Einstellung **INFO_LVL**.

```
BEGIN_EVENT_INFO
TIME_tvsec=1272348909
TIME_tvnsec=482502597
SEQUENCE_NUM=0
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=PROCESS_DOWN
END_EVPROD_INFO
END_EVENT_INFO
```

inetsock

Der Ereigniserzeuger **inetsock** überwacht TCP- (Transmission Control Protocol) und UDP-Socketoperationen (User Datagram Protocol).

Übersicht

Der Ereigniserzeuger **inetsock** wird im Verzeichnis "net" gespeichert und überwacht Socketoperationen.

Die folgenden Socketoperationen werden für TCP überwacht:

- Socket erstellen
- Port oder Adresse an den Socket binden
- Empfangsbereitschaft am Socket
- Verbindung am Socket annehmen und herstellen
- Verbindung zu einem Socket herstellen
- Verbindung zu einem Socket trennen
- Socket schließen

Die folgenden Socketoperationen werden für UDP überwacht:

- Socket erstellen
- Port oder Adresse an den Socket binden
- Socket schließen

Funktionen

```
AHAFS_THRESHOLD_STATE
AHAFS_CALLBACK_INTRCNTX
AHAFS_REMOTE_EVENT_ENABLED
```

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt Informationen, die im Protokollsteuerblock und am Socket verfügbar sind, im Rahmen seiner Ereignisdaten.

Die folgenden Daten werden für die TCP-Socketoperationen übergeben:

| Socketoperation | Daten |
|-----------------------|--|
| PRU_ATTACH | Allgemeine Informationen:
<ul style="list-style-type: none"> • PROG_NAME • SO_FAMILY • SO_PID • SO_PROTO • SO_TYPE • SO_UID |
| PRU_BIND | Allgemeine Informationen und die folgenden Elemente:
<ul style="list-style-type: none"> • SO_LADDR • SO_LPORT |
| PRU_LISTEN | Allgemeine Informationen und die folgenden Elemente:
<ul style="list-style-type: none"> • SO_LADDR • SO_LPORT |
| PRU_ACCEPT | Allgemeine Informationen und die folgenden Elemente:
<ul style="list-style-type: none"> • SO_FADDR • SO_FPORT • SO_LADDR • SO_LPORT |
| PRU_CONNECT | Allgemeine Informationen und die folgenden Elemente:
<ul style="list-style-type: none"> • SO_FADDR • SO_FPORT • SO_LADDR • SO_LPORT |
| PRU_DISCONNECT | Allgemeine Informationen und die folgenden Elemente:
<ul style="list-style-type: none"> • SO_FADDR • SO_FPORT • SO_LADDR • SO_LPORT |
| PRU_DETACH, PRU_ABORT | Allgemeine Informationen und die folgenden Elemente:
<ul style="list-style-type: none"> • SO_LADDR • SO_LPORT • SO_FADDR • SO_FPORT |

Die folgenden Daten werden für die UDP-Socketoperationen übergeben:

| Socketoperation | Daten |
|-----------------------|--|
| PRU_ATTACH | Allgemeine Informationen:
<ul style="list-style-type: none"> • PROG_NAME • SO_FAMILY • SO_PID • SO_PROTO • SO_TYPE • SO_UID |
| PRU_BIND, PRU_DYNBIND | Allgemeine Informationen und die folgenden Elemente:
<ul style="list-style-type: none"> • SO_LADDR • SO_LPORT |
| PRU_DETACH, PRU_ABORT | Allgemeine Informationen und die folgenden Elemente:
<ul style="list-style-type: none"> • SO_LADDR • SO_LPORT |

Zulässige Monitordateien

Zum Überwachen der Socketoperationen muss eine Monitordatei, die den Namen der zu überwachenden Socketoperationen hat, im Verzeichnis `inetsock.monFactory` erstellt werden. Wenn Sie beispielsweise die Erstellung von TCP-Sockets überwachen möchten, wird die Monitordatei `/aha/net/inetsock.monFactory/streamCreate.mon` verwendet. Für die Überwachung der Erstellung von UDP-Sockets wird die Datei `/aha/net/inetsock.monFactory/dgramCreate.mon` verwendet.

Die folgenden Dateien werden für alle mit AHAFS (Autonomic Health Advisor File System) überwachbaren TCP-Socketoperationen verwendet:

- `/aha/net/inetsock.monFactory/streamCreate.mon`
- `/aha/net/inetsock.monFactory/streamBind.mon`
- `/aha/net/inetsock.monFactory/streamListen.mon`
- `/aha/net/inetsock.monFactory/streamAccept.mon`
- `/aha/net/inetsock.monFactory/streamConnect.mon`
- `/aha/net/inetsock.monFactory/streamDisconnect.mon`
- `/aha/net/inetsock.monFactory/streamClose.mon`

Die folgenden Dateien werden für alle mit AHAFS überwachbaren UDP-Socketoperationen verwendet:

- `/aha/net/inetsock.monFactory/dgramCreate.mon`
- `/aha/net/inetsock.monFactory/dgramBind.mon`
- `/aha/net/inetsock.monFactory/dgramClose.mon`

Beispielereignisdaten

Die folgenden Ereignisdaten wurden von einem Prozess bei der Erstellung eines Sockets generiert. Dieses Beispiel ist die Ausgabe, die angezeigt wird, wenn die Ausgabebene 2 (INFO_LVL=2) ausgewählt ist:

```
AHAFS event: /aha/net/inetsock.monFactory/streamCreate.mon
```

```
-----  
BEGIN_EVENT_INFO  
Time       : Mon Jan 23 23:04:06 2012  
Sequence Num: 1  
RC_FROM_EVPROD=0  
BEGIN_EVPROD_INFO  
PROG_NAME=xmtopas  
SO_FAMILY=2  
SO_TYPE=1  
SO_PROTO=6  
SO_UID=0  
SO_PID=5243048  
END_EVPROD_INFO  
END_EVENT_INFO
```

Clusterereignisse

Wenn ein System zu einem Cluster gehört, kann es Benachrichtigungen über Ereignisse empfangen, die auf anderen Knoten eintreten, die zu demselben Cluster gehören. Ereigniserzeuger, in denen die Funktion **AHAFS_REMOTE_EVENT_ENABLED** angegeben ist, unterstützen die clusterweite Überwachung. Alle Ereigniserzeuger von AIX Event Infrastructure mit Ausnahme von **pidProcessMon** und **diskState** können solche fernen Benachrichtigungen bereitstellen.

Verhalten des Befehls **mkcluster** mit AIX Event Infrastructure:

Wenn AIX Event Infrastructure auf einem System nicht geladen ist und der Befehl **mkcluster** ausgeführt wird, wird das Pseudodateisystem von AIX Event Infrastructure an das Verzeichnis **/aha** angehängt und die Monitordateien werden über das Verzeichnis **/aha** gestartet. Wenn AIX Event Infrastructure bereits auf einem System geladen ist und der Befehl **mkcluster** ausgeführt wird, wird das Pseudodateisystem von AIX Event Infrastructure nicht erneut angehängt, und die Monitordateien werden über das Verzeichnis gestartet, an das das Pseudodateisystem von AIX Event Infrastructure angehängt wurde. Konsumenten Anwendungen müssen prüfen, wo das Pseudodateisystem von AIX Event Infrastructure angehängt wurde, um die Pfade der Monitordateien abzurufen.

Für den Empfang von Clusterereignissen müssen Konsumentenprozesse **CLUSTER=YES** angeben, wenn sie in die Monitordatei schreiben, die das Ereignis darstellt, um den Cluster überwachen zu können. Damit die fernen Ereignisse erkannt werden, muss ein Konsumentenprozess das Ereignis auf jedem Knoten überwachen, auf dem **CLUSTER=YES** angegeben ist.

Von einem fernen Knoten empfangene Ereignisse enthalten weder Benutzer- oder Prozessinformationen noch einen Stack-Trace, selbst wenn dies vom Ereigniserzeuger unterstützt wird.

Für Ereignisse, die auf einem fernen Knoten empfangen werden, wird kein Stack-Trace bereitgestellt, selbst wenn der Ereigniserzeuger dies unterstützt.

Die Clusterinformationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID** werden für alle Clustereignisse zwischen den Begrenzern **BEGIN_EVPROD_INFO** und **END_EVPROD_INFO** bereitgestellt. Anhand dieser Informationen kann das Überwachungsprogramm bestimmen, auf welchem Knoten das Ereignis eingetreten ist. Die in der Ausgabe des Befehls **lscluster -m** in den Feldern für die Clusterkurz-ID für den Knoten, für die **UIID** des Knotens und für die **UIIDs** des Clusters zurückgegebenen Informationen werden in der Ereignisausgabe von AIX Event Infrastructure in den Feldern **NODE_NUMBER**, **NODE_ID** bzw. **CLUSTER_ID** zurückgegeben.

Im Folgenden sehen Sie Beispielausgaben für ein lokales und ein fernes Vorkommen eines Ereignisses mit der **INFO_LVL**-Einstellung 2 und einem Ereigniserzeuger, in dem die Funktion **AHAFS_STKTRACE_AVAILABLE** definiert ist.

| Lokale Ereignisdaten | Ferne Ereignisdaten |
|--|---|
| BEGIN_EVENT_INFO
TIME_tvsec=1262670289
TIME_tvnssec=453840229
SEQUENCE_NUM=0
PID=4194474
UID=0
UID_LOGIN=0
GID=0
PROG_NAME=rpc.statd
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
NODE_NUMBER=1
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
EVENT_TYPE=PROCESS_DOWN
END_EVPROD_INFO
END_EVENT_INFO | BEGIN_EVENT_INFO
TIME_tvsec=1262670289
TIME_tvnssec=248144872
SEQUENCE_NUM=0
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=PROCESS_DOWN
NODE_NUMBER=1
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO |

Vordefinierte Ereigniserzeuger für eine clustersensitive AIX-Instanz

Diese Ereigniserzeuger sind nur verfügbar, wenn das System zu einem aktiven Cluster gehört.

nodeList

Der Ereigniserzeuger **nodeList** überwacht Änderungen der Clusterzugehörigkeit.

Übersicht

Der Ereigniserzeuger **nodeList** befindet sich im Verzeichnis "cluster" und überwacht Ereignisse,

bei denen Knoten im Cluster hinzugefügt oder entfernt werden. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört. Dieses Ereignis wird generiert, wenn ein Knoten im Cluster hinzugefügt oder entfernt wird (z. B. mit dem Befehl **chcluster**).

Funktionen

AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX

Rückkehrcodes

nodeList gibt 0 als Rückkehrcode zurück. **AHAFS_CLUSTER_REMOVE (-1)** wird nur dann zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **NODE_ADD** und **NODE_DELETE** im Rahmen seiner Ereignisdaten. Da er ein Erzeuger von Clusterereignissen ist, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Änderungen in der Knotenliste muss eine Monitordatei im Verzeichnis **nodeList.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname `/aha/cluster/nodeList.monFactory/nodeListEvent.mon`

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **nodeList**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271922590
TIME_tvnsec=886742634
SEQUENCE_NUM=1
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=NODE_ADD
NODE_NUMBER=1
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO
```

cIDiskList

Der Ereigniserzeuger **cIDiskList** überwacht Änderungen der Clusterzugehörigkeit.

Übersicht

Der Ereigniserzeuger **cIDiskList** befindet sich im Verzeichnis "disk" und überwacht Ereignisse, bei denen Platten im Cluster hinzugefügt oder entfernt werden. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört. Dieses Ereignis wird generiert, wenn eine Platte im Cluster hinzugefügt oder entfernt wird (z. B. mit dem Befehl **chcluster**).

Funktionen

AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX

Rückkehrcodes

cIDiskList gibt 0 als Rückkehrcode zurück. **AHAFS_CLUSTER_REMOVE (-1)** wird nur dann zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **DISK_ADD** und **DISK_DELETE** im Rahmen seiner Ereignisdaten im Feld **EVENT_TYPE**. **DISK_NAME** und **DISK_UID** der betroffenen Platte

werden ebenfalls übergeben. Da er zu den Erzeugern von Clusterereignissen gehört, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Änderungen in der Plattenliste muss eine Monitordatei im Verzeichnis **clDiskList.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname

```
/aha/disk/clDiskList.monFactory/clDiskListEvent.mon
```

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **clDiskList**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271927983
TIME_tvnsec=696543410
SEQUENCE_NUM=0
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=DISK_ADD
DISK_NAME=cldisk1
DISK_UID=3E213600A0B800016726C00000FF4B8677C80F1724-100 FAST03IBMfcp
NODE_NUMBER=2
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO
```

linkedCl

Der Ereigniserzeuger **linkedCl** wird generiert, wenn ein Cluster mit einem anderen Cluster verbunden wird bzw. wenn die Verbindung eines Clusters mit einem anderen Cluster aufgehoben wird.

Übersicht

Der Ereigniserzeuger **linkedCl** befindet sich im Verzeichnis "cluster" und überwacht Änderungen des Verbindungsstatus. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört. Dieses Ereignis wird generiert, wenn ein Cluster mit einem anderen Cluster verbunden wird bzw. wenn die Verbindung eines Clusters mit einem anderen Cluster aufgehoben wird.

Funktionen

```
AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX
```

Rückkehrcodes

linkedCl gibt 0 als Rückkehrcode zurück. **AHAFS_CLUSTER_REMOVE (-1)** wird nur dann zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **LINK_UP** und **LINK_DOWN** im Rahmen seiner Ereignisdaten. Er übergibt die Information **LINK_ID**. Da er ein Erzeuger von Clusterereignissen ist, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Änderungen in der Knotenliste muss eine Monitordatei im Verzeichnis **linkedCl.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname

```
/aha/cluster/linkedCl.monFactory/linkedClEvent.mon
```

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **linkedCI**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271224025
TIME_tvnsec=795042625
SEQUENCE_NUM=0
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=LINK_DOWN
LINK_ID=0x7BE9C1BD
NODE_NUMBER=1
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B0888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO
```

nodeContact

Der Ereigniserzeuger **nodeContact** überwacht den letzten Kontaktstatus des Knotens in einem Cluster.

Übersicht

Der Ereigniserzeuger **nodeContact** befindet sich im Verzeichnis "cluster" und überwacht den letzten Kontaktstatus des Knotens im Cluster. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört.

Funktionen

```
AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX
```

Rückkehrcodes

nodeContact gibt 0 als Rückkehrcode zurück. **AHAFS_CLUSTER_REMOVE (-1)** wird nur dann zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **CONNECT_UP** und **CONNECT_DOWN** im Rahmen seiner Ereignisdaten. Er übergibt den betroffenen **INTERFACE_NAME**. Da er ein Erzeuger von Clusterereignissen ist, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Änderungen in der Knotenliste muss eine Monitordatei im Verzeichnis **nodeContact.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname `/aha/cluster/nodeContact.monFactory/nodeContactEvent.mon`

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **nodeContact**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271921874
TIME_tvnsec=666770128
SEQUENCE_NUM=0
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=CONNECT_DOWN
INTERFACE_NAME=en0
NODE_NUMBER=2
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B0888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO
```

nodeState

Der Ereigniserzeuger **nodeState** überwacht den Status eines Knotens im Cluster.

Übersicht

Der Ereigniserzeuger **nodeState** befindet sich im Verzeichnis "cluster" und überwacht den Status eines Knotens im Cluster. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört. Dieses Ereignis wird beispielsweise generiert, wenn ein Knoten abstürzt oder beendet wird.

Funktionen

```
AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX
```

Rückkehrcodes

nodeState gibt 0 als Rückkehrcode zurück. **AHAFS_CLUSTER_REMOVE (-1)** wird nur dann zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **NODE_UP** und **NODE_DOWN** im Rahmen seiner Ereignisdaten. Da er ein Erzeuger von Clusterereignissen ist, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Änderungen des Status von Knoten muss eine Monitordatei im Verzeichnis **nodeState.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname `/aha/cluster/nodeState.monFactory/nodeStateEvent.mon`

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **nodeState**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271921536
TIME_tvnsec=68254861
SEQUENCE_NUM=1
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=NODE_UP
NODE_NUMBER=2
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO
```

nodeAddress

Der Ereigniserzeuger **nodeAddress** überwacht die Netzadresse des Knotens.

Übersicht

Der Ereigniserzeuger **nodeAddress** befindet sich im Verzeichnis "cluster" und überwacht die Netzadresse des Knotens. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört. Dieses Ereignis wird beispielsweise generiert, wenn ein Alias in einer Netzschnittstelle hinzugefügt oder entfernt wird.

Funktionen

```
AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX
```


Rückkehrcodes

nodeAddress gibt 0 als Rückkehrcode zurück. **AHAFS_CLUSTER_REMOVE (-1)** wird nur dann zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **ADDRESS_ADD** und **ADDRESS_DELETE** im Rahmen seiner Ereignisdaten. Er übergibt den Wert für **INTERFACE_NAME** der betroffenen Schnittstelle sowie die Werte für **FAMILY**, **ADDRESS** und **NETMASK** der IP-Adresse. Da er ein Erzeuger von Clusterereignissen ist, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Änderungen in der Knotenliste muss eine Monitordatei im Verzeichnis **nodeAddress.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname

```
/aha/cluster/nodeAddress.monFactory/nodeAddressEvent.mon
```

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **nodeAddress**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271922254
TIME_tvnsec=9053410
SEQUENCE_NUM=0
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=ADDRESS_ADD
INTERFACE_NAME=et0
FAMILY=2
ADDRESS=0x0A0A0A0A
NETMASK=0xFF000000
NODE_NUMBER=2
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO
```

networkAdapterState

Der Ereigniserzeuger **networkAdapterState** überwacht die Netzchnittstelle eines Knotens im Cluster.

Übersicht

Der Ereigniserzeuger **networkAdapterState** befindet sich im Verzeichnis "cluster" und überwacht die Netzchnittstelle eines Knotens im Cluster. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört. Dieses Ereignis wird generiert, wenn eine Netzchnittstelle inaktiviert oder aktiviert wird.

Funktionen

```
AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX
```

Rückkehrcodes

networkAdapterState gibt 0 als Rückkehrcode zurück. **AHAFS_CLUSTER_REMOVE (-1)** wird nur dann zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **ADAPTER_UP**, **ADAPTER_DOWN**, **ADAPTER_ADD** und **ADAPTER_DEL** im Rahmen seiner Ereignisdaten. Er übergibt den betroffenen **INTERFACE_NAME**. Da er ein Erzeuger von Clusterereignissen ist, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Änderungen in der Knotenliste muss eine Monitordatei im Verzeichnis **networkAdapterState.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname `/aha/cluster/networkAdapterState.monFactory/networkAdapterStateEvent.mon`

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **networkAdapterState**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271920539
TIME_tvnsec=399378269
SEQUENCE_NUM=1
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=ADAPTER_UP
INTERFACE_NAME=en0
NODE_NUMBER=2
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO
```

cIDiskState

Der Ereigniserzeuger **cIDiskState** überwacht Clusterplatten.

Übersicht

Der Ereigniserzeuger **cIDiskState** befindet sich im Verzeichnis "disk" und überwacht Clusterplatten. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört. Dieses Ereignis wird generiert, wenn eine Clusterplatte inaktiviert oder aktiviert wird.

Funktionen

```
AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX
```

Rückkehrcodes

cIDiskState gibt 0 als Rückkehrcode zurück. **AHAFS_CLUSTER_REMOVE (-1)** wird nur dann zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **DISK_UP** und **DISK_DOWN** im Rahmen seiner Ereignisdaten im Feld **EVENT_TYPE** zusammen mit dem Namen der betroffenen Clusterplatte. Da er ein Erzeuger von Clusterereignissen ist, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Clusterplatten muss eine Monitordatei im Verzeichnis **cIDiskState.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname `/aha/disk/cIDiskState.monFactory/cIDiskStateEvent.mon`

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **cIDiskState**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271935734
TIME_tvnsec=265210314
SEQUENCE_NUM=1
```

```

RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=DISK_DOWN
DISK_NAME=c1disk1
NODE_NUMBER=2
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO

```

repDiskState

Der Ereigniserzeuger **repDiskState** überwacht Repository-Platten.

Übersicht

Der Ereigniserzeuger **repDiskState** befindet sich im Verzeichnis "disk" und überwacht Repository-Platten. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört. Dieses Ereignis wird generiert, wenn eine Repository-Platte inaktiviert oder aktiviert wird.

Funktionen

```

AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX

```

Rückkehrcodes

repDiskState gibt 0 als Rückkehrcode zurück. **AHAFS_CLUSTER_REMOVE (-1)** wird nur dann zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **REP_UP** und **REP_DOWN** im Rahmen seiner Ereignisdaten im Feld **EVENT_TYPE** zusammen mit dem Plattennamen der betroffenen Repository-Platte. Da er ein Erzeuger von Clusterereignissen ist, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Repository-Platten muss eine Monitordatei im Verzeichnis **repDiskState.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname
/aha/disk/ repDiskState.monFactory/repDiskStateEvent.mon

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **repDiskState**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```

BEGIN_EVENT_INFO
TIME_tvsec=1271933757
TIME_tvnsec=134003703
SEQUENCE_NUM=1
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=REP_UP
DISK_NAME=hdisk2
NODE_NUMBER=2
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO

```

diskState

Der Ereigniserzeuger **diskState** überwacht Änderungen bei lokalen Platten.

Übersicht

Der Ereigniserzeuger **diskState** befindet sich im Verzeichnis "disk" und überwacht Änderungen bei lokalen Platten. Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster

gehört. Dieses Ereignis wird generiert, wenn eine lokale Platte inaktiviert oder aktiviert wird. Dieses Ereignis wird nur für Platten gemeldet, die vom Speicherframework unterstützt werden.

Funktionen

AHAFS_THRESHOLD_STATE
AHAFS_CALLBACK_INTRCNTX

Rückkehrcodes

diskState gibt 0 als Rückkehrcode zurück. Der Rückkehrcode **AHAFS_CLUSTER_REMOVE** (-1) wird nur zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **LOCAL_UP** und **LOCAL_DOWN** zusammen mit dem Namen der betroffenen lokalen Platte im Rahmen seiner Ereignisdaten. Als Erzeuger von Clusterereignissen übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung lokaler Platten muss eine Monitordatei im Verzeichnis **diskState.monFactory** erstellt werden. Der Name der Monitordatei muss im Format

```
/aha/disk/diskState.monFactory/<hdiskn>.mon
```

mit dem Namen der zu überwachenden lokalen Platte angegeben werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **diskState**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO  
TIME_tvsec=1271935029  
TIME_tvnsec=958362343  
SEQUENCE_NUM=1  
RC_FROM_EVPROD=0  
BEGIN_EVPROD_INFO  
EVENT_TYPE=LOCAL_UP  
DISK_NAME=hdisk4  
NODE_NUMBER=2  
NODE_ID=0xF079E8C801C11DF  
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404  
END_EVPROD_INFO  
END_EVENT_INFO
```

vgState

Der Ereigniserzeuger **vgstate** kann den Status der Datenträgergruppe auf einer Platte prüfen.

Übersicht

Der Ereigniserzeuger **vgState** befindet sich im Verzeichnis "disk". Dieser Ereigniserzeuger ist nur verfügbar, wenn das System zu einem Cluster gehört. Wenn eine lokale (bei **diskState**registrierte) Platte oder eine Clusterplatte aktiviert oder inaktiviert wird, wird ein entsprechendes **VG_UP**- bzw. **VG_DOWN**-Ereignis für die Datenträgergruppe auf dieser Platte ausgelöst. Mit diesem Ereigniserzeuger kann eine Anwendung den Status einer Datenträgergruppe auf der Platte mit dem LVM-Subsystem prüfen.

Funktionen

AHAFS_THRESHOLD_STATE
AHAFS_REMOTE_EVENT_ENABLED
AHAFS_CALLBACK_INTRCNTX

Rückkehrcodes

vgState gibt 0 als Rückkehrcode zurück. Der Rückkehrcode **AHAFS_CLUSTER_REMOVE** (-1) wird nur zurückgegeben, wenn der Cluster entfernt wird.

Nachrichten des Ereigniserzeugers

Dieser Ereigniserzeuger übergibt die Nachrichten **VG_UP** und **VG_DOWN** im Rahmen seiner Ereignisdaten. Außerdem übergibt er den Namen der betroffenen Platte und den Namen der Datenträgergruppe. Da er ein Erzeuger von Clusterereignissen ist, übergibt er zusätzlich die Informationen **NODE_NUMBER**, **NODE_ID** und **CLUSTER_ID**.

Zulässige Monitordateien

Für die Überwachung von Änderungen in der Knotenliste muss eine Monitordatei im Verzeichnis **vgState.monFactory** erstellt werden. Für die Monitordatei muss der Dateiname

```
/aha/disk/vgState.monFactory/vgStateEvent.mon
```

verwendet werden. In diesem Verzeichnis können keine anderen Monitordateien erstellt werden.

Beispielereignisdaten

Im Folgenden sehen Sie Ereignisdaten eines **vgstate**-Ereignisses mit der **INFO_LVL**-Standardeinstellung.

```
BEGIN_EVENT_INFO
TIME_tvsec=1271915408
TIME_tvnsec=699408296
SEQUENCE_NUM=0
RC_FROM_EVPROD=0
BEGIN_EVPROD_INFO
EVENT_TYPE=VG_UP
DISK_NAME=hdisk3
VG_NAME=myvg
NODE_NUMBER=2
NODE_ID=0xF079E8C801C11DF
CLUSTER_ID=0x6EA7B08888D811DFB918BEB25635B404
END_EVPROD_INFO
END_EVENT_INFO
```

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

*IBM Director of Licensing
IBM Europe, Middle East & Africa
Tour Descartes
2, avenue Gambetta
92066 Paris La Defense
France*

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuausgabe veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

*IBM Director of Licensing
IBM Europe, Middle East & Africa
Tour Descartes
2, avenue Gambetta
92066 Paris La Defense
France*

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des in diesen Informationen beschriebenen Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle genannten Leistungsdaten und Clientbeispiele dienen nur zur Veranschaulichung. Die tatsächlichen Leistungsdaten können je nach Konfiguration und Betriebsbedingungen variieren.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Alle von IBM angegebenen Preise sind empfohlene Richtpreise und können jederzeit ohne weitere Mitteilung geändert werden. Händlerpreise können u. U. von den hier genannten Preisen abweichen.

Diese Veröffentlichung dient nur zu Planungszwecken. Die in dieser Veröffentlichung enthaltenen Informationen können geändert werden, bevor die beschriebenen Produkte verfügbar sind.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufs. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren und können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind und Programmier Techniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Beispielprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Die Beispielprogramme werden ohne Wartung (auf "as-is"-Basis) und ohne jegliche Gewährleistung zur Verfügung gestellt. IBM übernimmt keine Haftung für Schäden, die durch die Verwendung der Beispielprogramme entstehen.

Kopien oder Teile der Beispielprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (Name Ihrer Firma) (Jahr).

Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corporation abgeleitet.

© Copyright IBM Corp. 2015.

Hinweise zur Datenschutzrichtlinie

IBM Softwareprodukte, einschließlich Software as a Service-Lösungen ("Softwareangebote"), können Cookies oder andere Technologien verwenden, um Informationen zur Produktnutzung zu erfassen, die Endbenutzererfahrung zu verbessern und Interaktionen mit dem Endbenutzer anzupassen oder zu anderen Zwecken. In vielen Fällen werden von den Softwareangeboten keine personenbezogenen Daten erfasst. Einige der IBM Softwareangebote können Sie jedoch bei der Erfassung personenbezogener Daten unterstützen. Wenn dieses Softwareangebot Cookies zur Erfassung personenbezogener Daten verwendet, sind nachfolgend nähere Informationen über die Verwendung von Cookies durch dieses Angebot zu finden.

Dieses Softwareangebot verwendet keine Cookies oder andere Technologien zur Erfassung personenbezogener Daten.

Wenn die für dieses Softwareangebot genutzten Konfigurationen Sie als Kunde in die Lage versetzen, personenbezogene Daten von Endbenutzern über Cookies und andere Technologien zu erfassen, müssen Sie sich zu allen gesetzlichen Bestimmungen in Bezug auf eine solche Datenerfassung, einschließlich aller Mitteilungspflichten und Zustimmungsanforderungen, rechtlich beraten lassen.

Weitere Informationen zur Nutzung verschiedener Technologien, einschließlich Cookies, für diese Zwecke finden Sie in der "IBM Online-Datenschutzerklärung, Schwerpunkte" unter <http://www.ibm.com/privacy>, in der "IBM Online-Datenschutzerklärung" unter <http://www.ibm.com/privacy/details> im Abschnitt "Cookies, Web-Beacons und sonstige Technologien" und unter "IBM Software Products and Software-as-a-Service Privacy Statement" unter <http://www.ibm.com/software/info/product-privacy>.

Marken

IBM, das IBM Logo und [ibm.com](http://www.ibm.com) sind Marken oder eingetragene Marken der IBM Corporation. Weitere Produkt- oder Servicenamen können Marken von IBM und anderen Herstellern sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite Copyright and trademark information unter www.ibm.com/legal/copytrade.shtml.

Linux ist eine Marke von Linus Torvalds in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

Windows ist eine Marke der Microsoft Corporation in den USA und/oder anderen Ländern.

Index

Sonderzeichen

@, integrierter Befehl
 C-Shell 300
:, integrierter Befehl
 Bourne-Shell 280
 Korn- bzw. POSIX-Shell 247
/, Stammdateisystem 457
. integrierter Befehl
 Korn- bzw. POSIX-Shell 247
. Integrierter Befehl
 Bourne-Shell 280
/dev/null, Datei 365
.env, Datei 335
/etc/hosts 19
/etc/inittab, Datei
 ändern 9
/etc/passwd, Datei 218
/etc/profile, Datei 55
/etc/security/passwd, Datei 312
/export, Verzeichnis 462
\$HOME, Verzeichnis 507
.hushlogin, Datei 626
.mwmrc, Datei 338
/opt, Dateisystem 455
~, (Ausgangs-)Verzeichnis 507
/proc, Dateisystem 455
.profile, Datei 55, 334
/usr/bin/ksh93 235
/usr/bin/psh, Befehl 218
/usr/bin/sh, Befehl 218
/usr/share, Verzeichnis 460
/var, Dateisystem 461
.Xdefaults, Datei 337
.xinitrc, Datei 336

Numerische Stichwörter

64-Bit-Modus
 Dateigruppen 57

A

Abbrechen
 Druckjobs 640, 641
 Vordergrundprozesse 149
Ablauf bei der Überwachung eines Ereignisses 657
Abmelden
 Übersicht 625
 vom Betriebssystem 626
Abrechnung der Verbindungsdauer 184
Abrechnungssystem
 Befehl runacct
 erneut starten 174
 starten 173
 Befehle
 automatische Ausführung 170
 über die Tastatur ausführen 171
 Übersicht 170
 Berichte
 Finanz- 169

Abrechnungssystem (*Forts.*)
 Berichte (*Forts.*)
 Monats- 167, 168
 Tages- 165
 Berichtsdaten
 Übersicht 165
 BSD-Systemmanager 348
 CPU-Belastung
 anzeigen 181
 Datei holidays
 aktualisieren 183
 Dateien
 Berichts- und Übersichtsdateien 173
 Dateien des Befehls runacct 174
 Datendateien 172
 Formate 176
 Übersicht 172
 Daten zur Druckerauslastung 169, 186
 anzeigen 182
 Daten zur Plattenbelegung 186
 anzeigen 182
 berichten 168
 Daten zur Systemaktivität
 anzeigen 179
 berichten 169
 während der Ausführung eines Befehls anzeigen 179
 Daten zur Verbindungsdauer
 anzeigen 182
 berichten 168
 erfassen 184
 Datensätze zusammenfassen 167
 Fehler
 Wiederherstellung 174
 Gebühren
 berechnen 187
 berichten 169
 konfigurieren 176
 Probleme
 nicht mehr auf dem neuesten Stand befindliche Datei
 holidays korrigieren 183
 runacct-Fehler beheben 190
 ungültige Dateiberechtigungen korrigieren 189
 ungültige Zeitangaben beheben 189
 Prozessdaten
 berichten 185
 erfassen 185
 tacct-Fehler
 beheben 187
 Übersicht 164
 wtmpt-Fehler
 beheben 188
Absoluter Pfadname 508
Abstrakte Sicht von AIX Event Infrastructure
 Definition 656
Abtrennen eines gespiegelten Datenträgers von einer Datenträgergruppe 395
ACL-Typ
 AIXC 319
 NFS4 320
acledit, Befehl 318, 325
aclget, Befehl 318, 324

- acldput, Befehl 318, 324
- ACLs 318
 - ACL-Beispiel für AIXC 321
 - Beispiel 324
 - für Dateisystemobjekte 318
 - verwalten 318
- Adressierbarkeit von Dateisystemfragmenten 494
- ahafs_evprods
 - Definition 656
- AIX
 - Übersicht für BSD-Systemmanager
 - Paging-Bereich 350
- AIX Event Infrastructure (AHAFS) 653
- AIX Runtime Expert 68
- aixterm, Befehl 332
- AIXwindows
 - Fenstermanager starten 336
 - Startdateien 336
- Aktivieren einer Datenträgergruppe (Vary-On) 371
 - Fehler beheben 407
- Aktivieren von Dateisystemen
 - zugeordneten Speicherbereich für Dateien mit Nullen auf-
füllen 496
- Aktivierte Dateisysteme
 - erstellen 496
 - freier Speicherplatz 496
 - große Dateien 496
- alias, Befehl 140
- alias, integrierter Befehl
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 253, 270
- Aliasing
 - Korn- bzw. POSIX-Shell 270
- Aliasnamen
 - auflisten 270
 - entfernen 270
 - erstellen 270
 - exportieren 270
 - mit absoluter Pfadangabe 271
 - nicht unterstützt 270
 - r 139
- Aliasnamen mit absoluter Pfadangabe 271
- Aliassubstitution
 - C-Shell 289
- Ändern
 - Benutzernamen 626
 - Berechtigungen 317
 - Desktop-Profile 632
 - Eingabeaufforderung 341
 - Priorität von Prozessen 148
 - Standardschriftart 340
 - Standardwerte 337
 - Steuertasten 340
- Anführungszeichen entfernen
 - Korn- bzw. POSIX-Shell 243
- Anhängen
 - Dateisystem 483
 - fern
 - Definition 483
 - lokal
 - Definition 483
 - mehrere Mounts verwenden 483
 - Übersicht 482
 - Workstation ohne Plattenspeicher anhängen
 - Sicherheit 485
- Anmeldedateien
 - /etc/environment 333
- Anmeldedateien (*Forts.*)
 - /etc/profile 334
 - Datei .env 335
 - Datei /etc/profile 55
 - Datei .profile 55, 334
- Anmelden
 - am Betriebssystem 625
 - mehrfach 626
 - mit einem anderen Benutzernamen 626
- Anmeldung
 - Benutzer-ID 312
 - Nachrichten unterdrücken 626
 - Name 624
 - Shell 218
 - Übersicht 625
 - Verzeichnisse 625
- Anpassen
 - Belegung der Maustasten 338
 - Bildschirmeinheiten 634
 - Farben und Schriftarten 337
 - Menüdefinitionen 338
 - Systemumgebung 339, 340, 341
 - Tastenbelegungen 338
- Anwendungsprogrammierschnittstelle
 - Workload Manager (API) 558
- Anzeigen
 - Anfangszeilen von Dateien anzeigen 209
 - angemeldete Benutzer 629
 - Anmeldename 627
 - Anzeige in Datei kopieren 367
 - Benutzer-ID 627
 - Dateiinhalte 205
 - Dateiinhalte eines Verzeichnisses 511
 - Dateitypen 205
 - Dateiverzeichnis 509
 - Gruppeninformationen 316
 - in Datei kopieren 368
 - Inhalte eines DOS-Verzeichnisses 216
 - Kalender 143
 - Konsolname 328
 - letzte Zeilen von Dateien 209
 - löschen 368
 - Name des Betriebssystems 628
 - seitenweise anzeigen 206
 - Softwareprodukte 330
 - Steuertastenkombinationen 330
 - Systemeinheiten 327
 - Systemname 629
 - Tastaturtabellen 330
 - Terminalname 329
 - Text am Bildschirm in großen Buchstaben 369
 - Text in großen Buchstaben anzeigen 369
 - Text seitenweise anzeigen 206
 - Umgebungsvariablen 331
 - verfügbare 329
 - verfügbare Schriftarten 329
 - verfügbarer Speicherplatz 464
 - Werte von Umgebungsvariablen 332
 - Zugriffssteuerungsinformationen 324
- API
 - Workload Manager (WLM) 558
- apropos, Befehl 145
- Arbeitsverzeichnis 507
- Argumente
 - in Befehlen 135
- Arithmetische Berechnungen
 - Korn- bzw. POSIX-Shell 228

- ASCII in PostScript
 - automatische Konvertierung 647, 648
 - Dateien konvertieren 647, 648
 - drucken 647
- ASCII-Terminals
 - hinzufügen 633
- at, Befehl 150, 151, 152
- atq, Befehl 151, 152
- Attribute
 - unterstützte Attribute in der Korn- bzw. POSIX-Shell 239
- Auflisten
 - Aliasnamen 270
 - geplante Prozesse 151
- Ausdrücke
 - Bedingung 231
 - C-Shell 305
 - Dateien mit Übereinstimmungen suchen 204
- Ausfall eines Plattenlaufwerks
 - Beispiel für Wiederherstellung 403
- Ausführen
 - Shell-Scripts 224
- Ausführungsebene
 - ermitteln 16
 - Protokoll anzeigen 16
- Ausgabe
 - in eine Datei umleiten 364
 - mit der Datei /dev/null verwerfen 365
 - Umleitung 363
 - Umleitungsoperator 363
- Auslagerungsspeicher
 - siehe Paging-Bereich 442
- Ausschneiden
 - Abschnitt von Textdateien 210
- Außer Kraft setzen
 - automatische Erkennung von Druckdateitypen 649
- Authentifizierung 312
- awk, Befehl 201

B

- backup, Befehl 25, 53
- Bänder
 - als Sicherungsdatenträger verwenden 25
 - Integrität prüfen 52
 - kopieren auf oder von 52
- Bandlaufwerke
 - Attribute
 - änderbar 601, 603, 604, 605, 606, 607, 608, 610, 611
 - Gerätedateien 612
 - verwalten 600
- banner, Befehl 369
- Batterie der Systemuhr 65
- Bearbeiten
 - Befehlsprotokoll 140
 - über die Befehlszeile in der Korn- bzw. POSIX-Shell 257
 - Zugriffssteuerungsinformationen 325
- Bedingte Substitution
 - Bourne-Shell 273
- Befehle 133
 - > 363
 - >> 364
 - < 365
 - /usr/bin/psh 218
 - /usr/bin/sh 218
 - | 366
 - acledit 318, 325
 - aclget 318, 324

- Befehle (*Forts.*)
 - acput 318, 324
 - aixterm 332
 - alias 140
 - at 150, 151, 152
 - atq 151, 152
 - awk 201
 - backup 25, 53
 - banner 369
 - Bourne-Shell 276
 - bsh 218, 223, 234, 271
 - C-Shell 299
 - capture 368
 - cat 206, 212, 364, 368
 - cd 507, 510
 - chdev 595
 - chfont 340
 - chgrp 324
 - chmod 315, 317, 318
 - chown 313, 324
 - chpq 648
 - clear 368
 - colrm 212
 - compress 40, 41
 - cp 203, 510
 - cpio 25
 - cpio -i 51
 - cpio -o 51
 - csd 218, 288
 - cut 210
 - date 66
 - Definition 219
 - del 214
 - df 464
 - diag 65
 - diff 208
 - dircmp 513
 - Direktaufrufnamen erstellen 140
 - dosdel 216
 - dosdir 216
 - dosread 215
 - doswrite 215
 - echo 368
 - Eingabe wiederholen 139
 - eingegabene Befehle speichern 137
 - env 331
 - exit 626
 - export 339
 - fdformat 49
 - file 205
 - find 53, 204
 - Flags 134
 - fcopy 50
 - format 49
 - fsck 24, 49
 - für BSD-Systemmanager 357
 - grep 11, 207, 366
 - Gruppen 313
 - head 209
 - history 137
 - id 313, 626, 627
 - in die Bourne-Shell integrierte 279
 - in die C-Shell integrierte 299, 300
 - integriert in die Korn- bzw. POSIX-Shell 246
 - kill 11, 152, 162
 - kombinieren 133
 - Korn- bzw. POSIX-Shell 264

Befehle (Forts.)

- ksh 51, 218, 267
- ln 212, 213, 505
- lock 326
- login 322, 626
- logname 627
- logout 626
- ls 313, 314, 511
- lsattr 595
- lscfg 327
- lscons 328
- lsdev 595
- lsdisp 329
- lsfont 329
- lsgroup 316
- lskbd 330
- lslpp 330
- man 136
- mkdev 595
- mkdir 508
- more 206
- mv 202
- mvdir 509
- mwm 336
- Namen 133
- nice 148
- nl 211
- pack 40, 42
- page 206
- Parameter 135
- passwd 630, 631
- paste 210
- pg 162, 206, 212
- pr 646
- printenv 332
- ps 11, 147, 162, 253
- psh 218, 267
- pwd 509
- qcan 640
- qchk 644
- qmov 642
- qpri 641
- qprt 637, 647
- r 139
- renice 148, 162
- restore 29, 53, 54
- rm 202, 214
- rmdir 513
- rsh 218, 223, 234
- script 368
- setclock 67
- sh 218
- shutdown 136
- sichern 53
- smit 29, 54, 340, 637, 641, 642, 645, 648
- smit rmat 152
- sort 207
- stty 330, 340
- su 322, 626
- Syntax 133
- tail 209
- tapechk 24, 52
- tar 25, 40, 52
- tcopy 52
- tee 367
- Textformatierung 141
- tn 11

Befehle (Forts.)

- touch 626
- tsh 218
- tty 329
- Übersicht 133
- uname 628, 629
- uncompress 40, 41, 42
- unpack 40, 42
- verketten 133
- Verwendungsanweisungen 135
- wc 208
- whatis 137
- whereis 136
- who 162, 627, 629
- who am i 627
- whoami 627
- xinit 336
- xlock 326
- zcat 42
- Zeichenfolgen ersetzen 139
- zusammengesetzte Befehle in der Korn-Shell 265
- Befehle und Direktaufrufe 376
- Befehls-Aliasing
 - Korn- bzw. POSIX-Shell 270
 - Tildensubstitution 271
- Befehls-Flags 133
- Befehlsketten
 - Definition 219, 366
- Befehlsliste
 - apropos 145
 - cal 143
 - factor 144
 - für Bourne-Shell 276
 - für C-Shell 297
 - für Korn- bzw. POSIX-Shell 230
- Befehlsprotokoll
 - bearbeiten 140
 - Korn- bzw. POSIX-Shell 269
 - Substitution 269
- Befehlssubstitution
 - Bourne-Shell 283
 - C-Shell 306
 - Korn- bzw. POSIX-Shell 227
- Befehlsübersichten 153
 - Anmeldenamen 631
 - Benutzerumgebung 332
 - Dateien 217
 - Dateisicherheit 326
 - Dateisysteme 514
 - drucken 649
 - E/A-Umleitung 369
 - Kennwörter 631
 - Sicherungsdateien 28
 - Speichermedien 28
 - System-IDs 631
 - Systeminformationen 332
 - Systemsicherheit 326
 - Verzeichnisse 514
- Befehlszeileneditierung
 - Editiermodus emacs 258
 - Editiermodus gmacs 258
 - Editiermodus vi 260, 261, 262, 263
 - Korn- bzw. POSIX-Shell 257
- Belegung der Maustasten 338
- Benannte Parameter 239
- Benutzer
 - angemeldete anzeigen 629

- Benutzer (*Forts.*)
 - Gruppen 313
 - Gruppeninformationen anzeigen 316
 - Klassen 314
 - System-ID anzeigen 627
 - wechseln 626
 - Benutzer-ID
 - anmelden 312
 - wechseln 626
 - Benutzerdefinierte Datenträgergruppen
 - importieren 472
 - Benutzerdefinierte Variablen 225
 - Bourne-Shell 284
 - Benutzerumgebungen
 - anpassen 55
 - Berechnungen
 - Primfaktoren, Zahlen 144
 - Berechtigung 322
 - Berechtigungen
 - Datei 317
 - Verzeichnisdateien 317
 - Betriebssystem
 - abmelden 626
 - anmelden 625
 - laden 13
 - Namen anzeigen 628
 - Systemabschluss durchführen 136
 - bg, integrierter Befehl
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 253
 - Bidirektionale Sprachen 332
 - Bildschirme
 - derzeit im System verfügbare auflisten 329
 - Bildschirminhalt löschen 368
 - Bildschirmnachrichten, reagieren 163
 - Binden eines Prozesses an einen Prozessor 161
 - Blöcke
 - Leistungseinbußen 494
 - Boot-Images
 - erstellen 14
 - Booten
 - abgestürztes System 6
 - BSD-Systemmanager 349
 - Informationen
 - RAM-Dateisystem 20
 - Systembootprozess 18
 - Übersicht 17
 - Wartungsmodus 19
 - Probleme diagnostizieren 22
 - Warmstart eines aktiven Systems durchführen 4
 - zu Wartungszwecken von der Festplatte 6
 - Bootprozess
 - Phasen 18
 - Bourne-Shell 218
 - bedingte Substitution 273
 - Befehle 276
 - Befehlssubstitution 283
 - benutzerdefinierte Variablen 284
 - Dateinamensubstitution 274
 - Ein- und Ausgabe umleiten 275
 - integrierte Befehle 279
 - Liste der integrierten Befehle 276
 - Mustererkennung 274
 - positionsgebundene Parameter 274
 - reservierte Wörter 278
 - Signalverarbeitung 278
 - starten 271
 - Bourne-Shell (*Forts.*)
 - Umgebung 272
 - Variablen 285
 - Variablensubstitution 284
 - vordefinierte Variablen 287
 - Zeichen in Anführungszeichen setzen 277
 - Zeichenklassen 222
 - zusammengesetzte Befehle 278
 - break, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 247
 - breaksw, integrierter Befehl
 - C-Shell 300
 - BSD 346, 347, 348, 355, 360, 362
 - Onlinedokumentation und Befehl man 346
 - Vergleich für Systemmanager 341, 342
 - Abrechnung 348
 - Befehle 357
 - booten und starten 349
 - cron 359
 - Dateien suchen und überprüfen 350
 - Dateisysteme 361
 - Dateivergleich 345
 - Drucker 355
 - Einheiten 359
 - Leistung 355
 - Netzbetrieb 342, 346, 351
 - NFS und NIS (früher Yellow Pages) 347
 - sichern 348
 - UUCP 360
 - Vergleich mit AIX für Systemmanager
 - Paging-Bereich 350
 - bsh, Befehl 218, 223, 234, 271
 - buf_wrap 669
 - Bytes
 - zählen 208
- ## C
- C-Shell 218
 - Aliassubstitution 289
 - Ausdrücke 305
 - Befehle 299
 - Befehlsausführung 307
 - Befehlssubstitution 306
 - Dateinamensubstitution 292
 - Ein- und Ausgabe umleiten 310
 - Einschränkungen 289
 - integrierte Befehle 299, 300
 - Jobsteuerung 296
 - Liste der integrierten Befehle 297
 - Operatoren 305
 - Protokolllisten 308
 - Protokollsubstitution 308
 - Signalverarbeitung 299
 - Startdateien 288
 - starten 288
 - Umgebungsvariablen 294
 - Variablensubstitution 290
 - Caching
 - Einschränkungen 616
 - Hinweise zur Hochverfügbarkeit 623
 - im dedizierten Modus konfigurieren 618
 - im NPIV-Modus konfigurieren 622
 - im virtuellen Modus konfigurieren 620
 - Komponenten 617

- Caching (*Forts.*)
 - konfigurieren 617
 - Konzept 615
 - Speicherdaten 615
 - Statistiküberwachung 623
 - verwalten 622
 - Vorteile 616
 - cal, Befehl 143
 - capture, Befehl 368
 - case, integrierter Befehl
 - C-Shell 300
 - cat, Befehl 206, 212, 364, 368
 - cd, Befehl 507, 510
 - cd, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 253
 - CD-ROM
 - Dateisysteme 468
 - CD-ROM File System (CDRFS) 488
 - CDPATH, Variable 225
 - CDRFS-Dateisysteme 469
 - cfgmgr 599
 - chdev, Befehl 595
 - chdir, integrierter Befehl
 - C-Shell 300
 - chfont, Befehl 340
 - chgrp, Befehl 324
 - chmod, Befehl 315, 317, 318
 - chown, Befehl 313, 324
 - chpq, Befehl 648
 - clDiskList 685
 - clDiskState 690
 - clear, Befehl 368
 - Clusterereignisse 683
 - Codierungsfiler 647
 - colrm, Befehl 212
 - COLUMNS, Variable 225
 - command, integrierter Befehl
 - Korn- bzw. POSIX-Shell 253
 - Common Desktop Environment 631
 - Bildschirme und Terminals 633
 - Bildschirme und Terminals hinzufügen 633
 - Bildschirmeinheiten anpassen 634
 - Profile ändern 632
 - compress, Befehl 40, 41
 - continue, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 247
 - cp, Befehl 203, 510
 - cpio, Befehl 25
 - cpio -i, Befehl 51
 - cpio -o, Befehl 51
 - CPU-Belastung
 - anzeigen 181
 - cron
 - für BSD-Systemmanager 359
 - cron, Dämon
 - Daten generieren 184
 - csh, Befehl 218, 288
 - cut, Befehl 210
- D**
- Dämonprozesse 145
 - date, Befehl 66
 - Datei
 - Baumstrukturen 453
 - Deskriptor 363, 365
 - Datei /etc/environment 333
 - Datei /etc/profile 334
 - Dateien 505
 - /dev/null 365
 - /etc/environment 333
 - /etc/passwd 218
 - /etc/profile 334
 - /etc/security/passwd 312
 - .hushlogin 626
 - .mwmrc 338
 - .profile 334
 - .Xdefaults 337
 - .xinitrc 336
 - Abgleichausdrücke 204
 - Abschnitte suchen 136
 - Anfangszeilen anzeigen 209
 - anhängen 483
 - archivieren 52
 - ASCII 197
 - ASCII auf einem PostScript-Drucker drucken 647
 - aus Anzeige kopieren 368
 - aus dem Speicher abrufen 52
 - ausführbar 197
 - ausgewählte Felder ausschneiden 210
 - bearbeiten 201
 - Berechtigungen 197, 314
 - Berechtigungen ändern 317
 - binär 197
 - Datei .env 335
 - DOS, löschen 216
 - durch Eingabeumleitung von der Tastatur erstellen 364
 - Eigentumsrecht 313
 - Eigentumsrecht ändern 313
 - eine Zeichenfolge suchen 207
 - einzelne Textzeilen anfügen 368
 - entfernen 202
 - entpacken 42
 - für Anzeige formatieren 205
 - für BSD-Systemmanager 345, 350
 - HISTFILE 269
 - in Ausgabe schreiben 209
 - Inhalt anzeigen 205
 - komprimieren 40
 - Komprimierung aufheben 42
 - kopieren 203
 - letzte Zeilen anzeigen 209
 - löschen 202
 - Metazeichen 200
 - Namenskonventionen 199
 - packen 40
 - Pfadnamen 199, 508
 - reguläre Ausdrücke 201
 - sichern 53
 - Spalten entfernen 212
 - Text einfügen 210
 - Text sortieren 207
 - Typ anzeigen 205
 - Typ ermitteln 205
 - Übersicht 197
 - umbenennen 202
 - Umgebung 333
 - verbinden 213
 - Verbindung 212
 - vergleichen 208, 513

- Dateien (*Forts.*)
 - verknüpfen 364
 - verknüpfte entfernen 214
 - verschieben 202
 - von Band oder Platte kopieren 51
 - von DOS kopieren 215
 - wiederherstellen 29, 33, 54
 - Zeilen aus mehreren Dateien zusammenfügen 210
 - Zeilen nummerieren 211
 - Zugriffsmodi 314
 - zum Drucken formatieren 646
- Dateinamenssubstitution
 - Bourne-Shell 274
 - C-Shell 292
 - Korn- bzw. POSIX-Shell 242
- Dateisystem
 - Images 503
 - umgehen 421
- Dateisysteme 505
 - /opt 455
 - /proc 455
 - abhängen 466
 - anhängen 466, 483
 - auf wieder beschreibbaren optischen Datenträgern 469
 - Befehle für die Verwaltung 463, 465
 - Beispiel 199
 - Benutzerdateisysteme sichern 28
 - beschädigte korrigieren 476
 - Beschreibung 624
 - CD-ROM File System (CDRFS) 488
 - CDRFS 468, 469
 - Dateibaum
 - /, Stammdateisystem 457
 - Dateisystem /usr 458
 - Dateisystem /var 461
 - Stammdateisystem (/) 457
 - Übersicht 454
 - Verzeichnis /export 462
 - Verzeichnis /usr/share 460
 - Dateien mit freien Bereichen 496
 - Datenkomprimierung 497
 - Enhanced Journaled File System (JFS2) 488
 - Fragmente 490
 - für BSD-Systemmanager 361
 - große Dateien 496
 - Größe in der Stammdatenträgergruppe verringern 472
 - Gruppen
 - abhängen 466
 - anhängen 466
 - home 455
 - I-Nodes 490
 - Integrität prüfen 470
 - interaktive Korrekturen vornehmen 49
 - Journaled File System (JFS) 453, 488
 - Konsistenz überprüfen 49
 - mit Scripts sichern 47
 - Network File System (NFS) 488
 - Plattenüberlauf 478
 - Stamm- 455
 - Struktur 455
 - Typen
 - CD-ROM 488
 - DVD-ROM 488
 - Enhanced Journaled File System (JFS2) 488
 - Journaled File System (JFS) 488
 - Network File System (NFS) 488
 - Übersicht 453
- Dateisysteme (*Forts.*)
 - UDFS 468
 - Verfahren zur Journalführung 453
 - verfügbarer Speicherplatz 464
 - Verwaltungstasks 463
- Dateisystemloser logischer Datenträger
 - definieren 421
- Dateisystemprotokoll 381
- Dateitypen
 - binär 197
 - Text 197
 - Verzeichnisdateien 197
- Daten von einer Platte ohne erneute Formatierung wiederherstellen 398
- Datenkomprimierung 497
 - Fragmente 490
 - Leistungseinbußen 500
- Datenträgergruppe
 - einen gespiegelten Datenträger abtrennen 395
 - spiegeln 419
 - Spiegelung entfernen 422
 - Stamm-
 - spiegeln 420
- Datenträgergruppen
 - Aktivieren einer Datenträgergruppe (Vary-On) 371
 - benutzerdefinierte
 - importieren 472
 - Definition 415
 - exportieren 383
 - gespiegelt
 - einen physischen Datenträger ersetzen 389
 - Gründe für das Erstellen separater Datenträgergruppen 428
 - hohe Verfügbarkeit 428
 - Implementierung der Richtlinie 442
 - importieren 383
 - in eine Datenträgergruppe ohne Quorum ändern 375
 - ohne Quorum 374
 - Platte austauschen 404
 - Quorum 372
 - Strategie 428
 - verschieben 383
- Datenträgergruppen ohne Quorum 374
- default, integrierter Befehl
 - C-Shell 300
- del, Befehl 214
- Deskriptorbereich der Datenträgergruppe (VGDA) 371
- df, Befehl 464
- diag, Befehl 65
- Diagnose von Bootproblemen
 - Warmstart eines Systems mit einem integrierten Grafiks subsystem 8
 - Zugriff auf ein System, das nicht bootet 21
- Diagnose von Problemen bei Plattenlaufwerken 396
- Diagnosenachricht 363
- diff, Befehl 208
- dircmp, Befehl 513
- Direktaufrufname für Befehle
 - erstellen 140
- dirs, integrierter Befehl
 - C-Shell 300
- Disketten
 - als Sicherungsdatenträger verwenden 25
 - formatieren 49
 - kopieren auf oder von 50
 - Umgang mit 25
- diskState 691

- DOS-Dateien
 - Inhalt auflisten 216
 - konvertieren 215
 - kopieren 215
 - löschen 216
- dosdel, Befehl 216
- dosdir, Befehl 216
- dosread, Befehl 215
- doswrite, Befehl 215
- Dreistellige Anzeige 625
- Drucken
 - ASCII-Dateien auf einem PostScript-Drucker 647
 - Dateien formatieren 646
 - Druckerstatus 644, 645
 - Druckjobs abbrechen 640, 641
 - Druckjobs Prioritäten zuweisen 641
 - Druckjobs starten 637
 - Druckjobs verschieben 642
 - Erkennung von Druckdateitypen außer Kraft setzen 649
 - Status von Druckjobs prüfen 645
 - Übersicht 637
- Drucker
 - für BSD-Systemmanager 355
 - Positionscodes 566
 - Status 645
- Druckerauslastungsabrechnung 186
- Druckjobs
 - abbrechen 640, 641
 - Priorität zuweisen 641
 - starten 637
 - Status prüfen 644, 645
 - verschieben 642
- Druckwarteschlange
 - Status 644, 645
- DVD
 - Dateisysteme 468
- Dynamische Prozessorfreigabe 58, 59

E

- E/A-Umleitung
 - Bourne-Shell 275
 - C-Shell 310
 - Korn- bzw. POSIX-Shell 243
 - Standard 363
- echo, Befehl 368
- echo, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 253
- ed, Editor 201
- EDITOR, Variable 225
- Editoren
 - Befehlszeileneditierung 257
 - ed 201
 - emacs 257, 258
 - gmacs 257, 258
 - vi 201, 257
- EFS
 - Encrypted File Systems 463
- Einfache Befehle
 - Definition 219
- Einfaches Netz 19
- Einfügen
 - Abschnitt von Textdateien 210
- Eingabe
 - Umleitung 363

- Eingabe (*Forts.*)
 - Umleitungsoperator 365
- Eingabeaufforderung
 - ändern 341
- Eingabemodus
 - Befehle für Eingabeeditierung 261
 - Definition 260
- Einheit
 - für BSD-Systemmanager 359
 - Installation 410
 - Laufwerk für wieder beschreibbare optische Platten konfigurieren 410
- Einheiten 411
 - Anschlüsse überprüfen 597
 - Attribute ändern 595
 - Attribute überprüfen 595
 - Bereitschaftsstatus überprüfen 598
 - Diagnose durchführen 599
 - große Anzahl konfigurieren 411
 - Informationen anzeigen 327
 - Klassen 563
 - Knoten 563
 - MPIO
 - verkabeln 578
 - MPIO-fähig 577
 - neue definieren 595
 - Positionscodes 565
 - Software überprüfen 595
 - Status 564
 - Status überprüfen 595
- Einheitenkonfigurationsdatenbank
 - mit Logical Volume Manager synchronisieren 406
- Einheitentreiber
 - Auswirkung der Fragmentverwendung auf Größe 503
- Einschränkungen
 - logische Datenträger 405
- Einzelbenutzermodus 55
- Einzelbenutzersysteme
 - Ausführungsebenen ändern 17
- else, integrierter Befehl
 - C-Shell 300
- emacs, Editor
 - Befehlszeileneditierung 257, 258
- end, integrierter Befehl
 - C-Shell 300
- endif, integrierter Befehl
 - C-Shell 300
- endsw, integrierter Befehl
 - C-Shell 300
- Enhanced Journaled File System (JFS2) 488
- Entfernen
 - Aliasnamen 270
 - Hintergrundprozesse 152
 - lokaler Bildschirm 633
 - Prozesse aus dem Zeitplan 152
 - Spalten in Textdateien 212
 - verknüpfte Dateien 214
- Entfernen von Einheiten im laufenden Betrieb 378, 423
- Entpacken
 - Dateien 42
- env, Befehl 331
- ENV, Variable 225
- Ereignisdaten lesen 664
- Ereigniserzeuger
 - Definition 656
- Ereigniskonsumenten 655
- Erneues Starten des Systems 13

- Erneut starten
 - gestoppte Prozesse 149
- ERRNO, Variable 224
- Erstellen
 - Aliasnamen 270
 - Befehlsalias 140
 - Shell-Scripts 224
 - Verzeichnisse 508
- Erweiterte Korn-Shell
 - arithmetische Erweiterungen 235
 - assoziative Arrays 235
 - Befehlsrückgabewerte 235
 - Beschreibung 235
 - Discipline-Funktionen 235
 - Funktionsumgebungen 235
 - integrierte Befehle 235
 - Parametererweiterungen 235
 - Referenzen auf Variablennamen 235
 - Shellprotokoll 235
 - Suchregeln für PATH 235
 - Variablen 235
 - Verbundvariablen 235
 - Zuweisung von Verbundvariablen 235
- eval, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 247
- exec, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 247, 267
- exit, Befehl 626
- exit, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 247
- Exit-Status
 - Korn- bzw. POSIX-Shell 238
- export, Befehl 339
- export, integrierter Befehl 246
 - Bourne-Shell 280
 - Korn- bzw. POSIX-Shell 247, 267
- Exportieren
 - Aliasnamen 270
 - Shellvariablen 339

F

- factor, Befehl 144
- fc, integrierter Befehl 140, 269
 - Korn- bzw. POSIX-Shell 253
- FCEDIT, Variable 225
- fdformat, Befehl 49
- Fehlerformat 668
- Fehlernachricht 363
- Fehlerprotokollierung
 - auf Einheitenfehler überprüfen 595
- Feldtrennung
 - Korn- bzw. POSIX-Shell 229
- Femote
 - Shell 218
- Fern
 - Anmeldung 624
- Festlegen
 - Anfangspriorität von Prozessen 148
 - Zugriffssteuerungsinformationen 324
- Festplatte 386

- Festplattenlaufwerke 478
 - siehe auch Plattenlaufwerke 397
- fg, integrierter Befehl
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 253
- file
 - Befehl 205
- Filter 366
- find, Befehl 53, 204
- Flags 133
 - für Befehl pr 646
 - für Befehl qprt 637, 647
 - in Befehlen 134
- flcopy, Befehl 50
- foreach, integrierter Befehl
 - C-Shell 300
- format, Befehl 49
- Formatieren
 - Dateien zum Drucken 646
 - Disketten 49
- FPATH, Variable 225
- Fragmente
 - Auswirkung auf Sicherung/Wiederherstellung 503
 - Auswirkungen auf die Plattenbelegung 490
 - Einschränkung für Einheitentreiber 503
 - Größe
 - angeben 492
 - ermitteln 492
 - Leistungseinbußen 494
 - und variable Anzahl von I-Nodes 490
- fsck, Befehl 24, 49
- Funktion für Entfernen von Einheiten im laufenden Betrieb 404, 423, 424

G

- Ganzzahlberechnungen 228
- Gebührenabrechnung 187
- Geisterprozesse 145
- Gespiegelte Datenträgergruppe
 - einen physischen Datenträger ersetzen 389
- getopts, integrierter Befehl
 - Korn- bzw. POSIX-Shell 253
- glob, integrierter Befehl
 - C-Shell 300
- gmacs, Editor
 - Befehlszeileneditierung 257, 258
- goto, integrierter Befehl
 - C-Shell 300
- grep, Befehl 11, 207, 366
- groups, Befehl 313

H

- hangups, integrierter Befehl
 - C-Shell 300
- hash, integrierter Befehl
 - Bourne-Shell 280
- hashstat, integrierter Befehl
 - C-Shell 300
- head, Befehl 209
- Here-Dokument 243, 366
- Hierarchisches Netz 19
- Hintergrundprozesse 145
- HISTFILE
 - Datei 269

HISTFILE (*Forts.*)
 Variable 225
 history
 Listen in der C-Shell 308
 history, Befehl 137
 history, integrierter Befehl
 C-Shell 300
 HISTSIZE, Variable 225, 269
 home, Dateisystem 455
 HOME, Variable 225
 home, Verzeichnis 507
 Hot-Plug-Verwaltung
 PCI 570
 Hot-Spots in logischen Datenträgern 440

I

I-Node-Nummer 197, 212, 505
 I-Node-Referenznummer 505
 I-Nodes 492
 Number of Bytes Per (NBPI)
 angeben 492
 ermitteln 492
 und Fragmente 490
 variable Anzahl 492
 I-Nodes, Anzahl 494
 id, Befehl 313, 626, 627
 idbgen 56
 IDs
 Benutzer 313
 if, integrierter Befehl
 C-Shell 300
 IFS, Variable 225
 Importieren benutzerdefinierter Datenträgergruppen 472
 Inaktives System
 Hardware überprüfen 10
 Prozesse überprüfen 11
 System erneut starten 13
 inetsock 681
 inittab, Datei 9
 Dämon srcmstr 194
 Inline-Eingabedokumente 366
 Integrierte Befehle
 : 247, 280
 . 247, 280
 @ 300
 alias 253, 270, 300
 Befehl 253
 bg 253, 300
 Bourne-Shell 276, 279
 break 247, 280, 300
 breaksw 300
 C-Shell 299, 300
 case 300
 cd 253, 280, 300
 chdir 300
 continue 247, 280, 300
 Definition 219
 dirs 300
 echo 253, 280, 300
 else 300
 end 300
 endif 300
 endsw 300
 eval 247, 280, 300
 exec 247, 267, 280, 300
 exit 247, 280, 300
 Integrierte Befehle (*Forts.*)
 export 246, 247, 267, 280
 fc 140, 253, 269
 fg 253, 300
 foreach 300
 getopts 253
 glob 300
 goto 300
 hangups 300
 hash 280
 hashstat 300
 history 300
 if 300
 jobs 253, 296, 300
 kill 253, 300
 Korn- bzw. POSIX-Shell 246
 let 228, 253
 limit 300
 login 300
 logout 300
 newgrp 247
 nice 300
 notify 300
 onintr 300
 popd 300
 print 253
 pushd 300
 pwd 253, 280
 read 253, 280, 283
 readonly 246, 247, 280
 reguläre Befehle 246, 253, 279
 rehash 300
 repeat 300
 return 247, 280
 set 247, 274, 280, 300
 setenv 300
 setgroups 253
 setsenv 253
 shift 247, 274, 280, 300
 Sonderbefehle 246, 247, 279, 280
 source 300
 Standard 300
 stop 300
 suspend 300
 switch 300
 test 253, 280
 time 300
 times 247, 280
 trap 247, 280
 type 280
 typeset 228, 239, 246, 247, 267
 ulimit 253, 280
 umask 253, 280, 300
 unalias 253, 270, 300
 unhash 300
 unlimit 300
 unset 247, 280, 300
 unsetenv 300
 wait 253, 280, 300
 whence 253
 while 300
 Integrierte Sonderbefehle
 Bourne-Shell 280
 Korn- bzw. POSIX-Shell 230, 247
 Interpretation
 Leerzeichen 288

J

- JFS (Journaled File System)
 - auf einen anderen physischen Datenträger kopieren 504
 - auf wieder beschreibbaren optischen Datenträgern 469
 - Datenkomprimierung 497
 - Fragmente 490
 - Größenbegrenzungen 493
 - maximale Größe 494
 - mit variabler Anzahl von I-Nodes 490
- JFS-Protokoll 381
 - Größe 494
- JFS2 (Enhanced Journaled File System)
 - Größenbegrenzungen 493, 495
- JFS2-Protokoll 381
- Jobs
 - aus dem Zeitplan entfernen 152
 - geplante auflisten 151
 - planen 150
- jobs, integrierter Befehl
 - C-Shell 296, 300
 - Korn- bzw. POSIX-Shell 253
- Jobsteuerung
 - C-Shell 296
 - Korn- bzw. POSIX-Shell 256
- Journaled File System (JFS) 453, 488

K

- Kabel
 - Anschlüsse überprüfen 597
- Kalender
 - anzeigen 143
- Kennung
 - Definition 219
- Kennwörter
 - ändern oder festlegen 630
 - außer Kraft setzen 631
 - Beschreibung 624
 - Richtlinien 630
- Kernelerweiterung für AIX Event Infrastructure 654
- kill, Befehl 11, 152, 162
- kill, integrierter Befehl
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 253
- Klassen
 - Benutzer 314
- Kombinieren von Befehlen 133
- Kommentare
 - Definition 219
- Komponenten von AIX Event Infrastructure
 - Definition 654, 664
- Komprimieren
 - Dateien 40
- Komprimierung aufheben
 - Dateien 41, 42
- Konsole
 - Namen anzeigen 328
- Konsolidierung doppelter Ereignisse 667
- Konvertieren
 - ASCII-Dateien in PostScript 647
 - DOS-Dateien 215
- Kopieren
 - Anzeige in Datei 368
 - auf oder von Band 52
 - auf oder von Disketten 50
 - Dateien 203

Kopieren (Forts.)

- Dateien auf Band oder Platte 51
- Dateien des Basisbetriebssystems 215
- Dateien von Band oder Diskette 51
- DOS-Dateien 215
- Koprozesse
 - Korn- bzw. POSIX-Shell 245
- Korn- bzw. POSIX-Shell 218
 - Anführungszeichen entfernen 243
 - Anführungszeichen verwenden 232
 - arithmetische Berechnungen 228
 - bearbeiten 257
 - Bedingungsausdrücke 231
 - Befehle verwenden 264
 - Befehls-Aliasing 270
 - Befehlsprotokoll 269
 - Befehlssubstitution 227
 - benutzerdefinierte Variablen 225
 - Dateinamenssubstitution 242
 - Ein- und Ausgabe umleiten 243
 - Eingabe und Ausgabe von Koprozessen umleiten 246
 - erweitert 235
 - Exit-Status 238
 - Feldtrennung 229
 - Funktionen 268
 - integrierte Befehle 246
 - Jobsteuerung 256
 - Koprozesse 245
 - Liste der integrierten Sonderbefehle 230
 - Liste der regulären integrierten Befehle 230
 - Mustererkennung 242
 - Parametersubstitution 239, 240
 - reservierte Wörter 234
 - Signalverarbeitung 257
 - starten 267
 - Tildensubstitution 271
 - Umgebung 267
 - vordefinierte Parameter 241
 - vordefinierte Variablen 224
 - zusammengesetzte Befehle 265
- ksh, Befehl 51, 218, 267
- ksh93
 - arithmetische Erweiterungen 235
 - assoziative Arrays 235
 - Befehlsrückgabewerte 235
 - Beschreibung 235
 - Discipline-Funktionen 235
 - FunktionMsumgebungen 235
 - integrierte Befehle 235
 - Parametererweiterungen 235
 - Referenzen auf Variablenamen 235
 - Shellprotokoll 235
 - Suchregeln für PATH 235
 - Variablen 235
 - Verbundvariablen 235
 - Zuweisung von Verbundvariablen 235

L

- LANG, Variable 225
- LC_ALL, Variable 225
- LC_COLLATE, Variable 225
- LC_CTYPE, Variable 225
- LC_MESSAGES, Variable 225
- Leerzeichen
 - Definition 219
 - Interpretation 288

- Leistung
 - BSD-Systemmanager 355
 - verbessern
 - logische Datenträger ohne Dateisystem definieren 421
 - Lesen der dreistelligen Anzeige 625
 - let, integrierter Befehl
 - Korn- bzw. POSIX-Shell 228, 253
 - limit, integrierter Befehl
 - C-Shell 300
 - LINENO, Variable 224
 - LINES, Variable 225
 - linkedCI 686
 - Listen
 - Definition 219
 - ln, Befehl 212, 213, 505
 - lock, Befehl 326
 - Logical-Volume Control Block
 - nicht geschützt vor Zugriff auf den logischen Datenträger 421
 - Logical Volume Manager 376
 - Logical Volume Manager (LVM) 409
 - Definition 371
 - mit der Einheitenkonfigurationsdatenbank synchronisieren 406
 - login
 - Namen anzeigen 627
 - login, Befehl 322, 626
 - login, integrierter Befehl
 - C-Shell 300
 - Logische Datenträger
 - Dateisystem auf einem neuen logischen Datenträger hinzufügen 463
 - Dateisysteme 417
 - Datenträgergruppen 415
 - Datenträgergruppen ohne Quorum 374
 - Definition 417
 - einheitenübergreifend 439
 - Einschränkungen 405
 - Größe
 - erhöhen 463
 - prüfen 463
 - verringern 463
 - Hot-Spots 440
 - Inhalt auf ein anderes System verschieben 384
 - Namen ändern 379
 - physische Datenträger 414
 - Planungsrichtlinie für Schreiboperationen 431, 432
 - Platte austauschen 404
 - Quorum 372
 - Richtlinie für Datenträgergruppen 442
 - Strategie 430
 - write-verify-Richtlinie 439
 - Zuordnungsdateien 438
 - Logische Partitionen
 - Definition 417
 - Größe definieren 472
 - Strategie für plattenübergreifende Zuordnung 434
 - Logischer Datenträger
 - auf einen anderen physischen Datenträger kopieren 379
 - ohne Dateisystem
 - definieren 421
 - logname, Befehl 627
 - logout, Befehl 626
 - logout, integrierter Befehl
 - C-Shell 300
 - Löschen
 - Dateien 202
 - Löschen (*Forts.*)
 - DOS-Dateien 216
 - Verzeichnisse 513
 - ls, Befehl 313, 314, 511
 - lsattr, Befehl 595
 - lscfg, Befehl 327
 - lscons, Befehl 328
 - lsdev, Befehl 595
 - lsdisp, Befehl 329
 - lsfont, Befehl 329
 - lsgroup, Befehl 316
 - lskbd, Befehl 330
 - lslpp, Befehl 330
 - lssrc, Befehl 196
 - LVCB (Logical-Volume Control Block)
 - nicht geschützt vor Zugriff auf den logischen Datenträger 421
 - LVM 376, 409
- ## M
- MAIL, Variable 225
 - MAILCHECK, Variable 225
 - MAILPATH, Variable 225
 - man, Befehl 136
 - BSD-Systemmanager 346
 - Man-Page
 - finden mit Schlüsselwortsuche 145
 - Mehrbenutzersysteme
 - Ausführungsebenen ändern 16
 - Menüdefinitionen 338
 - Metazeichen 200
 - Anführungszeichen in Korn- bzw. POSIX-Shell 232
 - Definition 219
 - Mit Nullen auffüllen, zugeordneten Speicherbereich für Dateien 496
 - mkdev, Befehl 595
 - mkdir, Befehl 508
 - moddir 673
 - modDir 681
 - modfile 670, 672
 - modFile 681
 - Monitordatei erstellen
 - Definition 658
 - more, Befehl 206
 - motd, Datei 67
 - Mount durchführen
 - /etc/filesystem 484
 - automatische Mounts 484
 - Workstation ohne Plattenspeicher anhängen
 - Beschreibung 487
 - Mountpunkte 483
 - MPIO 575
 - verwalten 577
 - Multi-Path I/O 575
 - Multiprotokollanschluss
 - Positionscodes 568
 - Mustererkennung
 - Bourne-Shell 274
 - Korn- bzw. POSIX-Shell 242
 - mv, Befehl 202
 - mmdir, Befehl 509
 - mwm, Befehl 336

N

Nachricht des Tages
 ändern 67
Nachrichten
 am Bildschirm anzeigen 368
 an Standardausgabe senden 368
Nachrichten am Bildschirm, reagieren 163
Namenskonventionen
 Dateien 199
 Verzeichnisse 507
NBPI 492
Network File System (NFS) 488
networkAdapterState 689
Netz
 für BSD-Systemmanager 342, 346, 351
 Systemnamen anzeigen 629
Netzplanung
 TCP/IP 19
newgrp, integrierter Befehl
 Korn- bzw. POSIX-Shell 247
NFS und NIS
 BSD-Systemmanager 347
nice, Befehl 148
nice, integrierter Befehl
 C-Shell 300
Nicht funktionsfähiges System
 Hardware überprüfen 10
 Prozesse überprüfen 11
 System erneut starten 13
Nicht grafikfähiges Terminal (TTY, Teletypewriter)
 Positionscodes 566
NIS 347
nl, Befehl 211
NLSPATH, Variable 225
nodeAddress 688
nodeContact 687
nodeList 684
nodeState 688
Notfall
 Systemabschluss 55
notify, integrierter Befehl
 C-Shell 300
NUM_EVDROPS_INTRCNTX 669
Number of Bytes Per I-Node (NBPI) 492
Nummerieren
 Zeilen in Textdateien 211

O

OLDPWD, Variable 224
onintr, integrierter Befehl
 C-Shell 300
Operanden
 in Befehlen 135
Operator für Umleiten mit Anfügen 364
Operatoren
 C-Shell 305
OPTARG, Variable 224
OPTIND, Variable 224
Optionen
 in Befehlen 134
Optischer Datenträger
 Dateisysteme auf wieder beschreibbaren verwenden 469
Optisches Laufwerk
 konfigurieren 410

P

pack, Befehl 40, 42
page, Befehl 206
Paging-Bereich
 AIX für BSD-Systemmanager 350
 Befehle für die Verwaltung 445
 entfernen 447
 Größe von hd6 ändern 448
 hd6 verschieben 448
 Merkmale ändern 447
 Merkmale für das Erstellen 445
 Übersicht 442
 zuordnen 443
 Zuordnungsmodus early 443
 Zuordnungsmodus late 443
Parameter
 benannt 239
 in Befehlen 135
 Korn- bzw. POSIX-Shell 239, 241
 positionsgebunden 239
 Sonderbefehle 241
 Sonderparameter 239
 vordefinierte 241
Parametersubstitution
 Korn- bzw. POSIX-Shell 240
Parameterzuordnungslisten
 Definition 219
passwd, Befehl 630, 631
paste, Befehl 210
PATH, Variable 225
Pfade
 Verzeichnisdateien 508
Pfadnamen
 absolut 199, 508
 Dateien 199
 relativ 508
 Verzeichnisdateien 508
pg, Befehl 162, 206, 212
Physische Datenträger
 aus einem verfügbaren Plattenlaufwerk erstellen 388
 Definition 414
 eine Platte konfigurieren 386
 einer gespiegelten Datenträgergruppe ersetzen 389
 Inhalt verschieben 384
Physische Partitionen
 Definition 416
 Größe 416
Physischer Datenträger
 JFS auf einen anderen kopieren 504
 logischen Datenträger kopieren 379
PID 145
pidProcessMon 680, 681
Pipes 366
Planen
 Prozesse 150
Planungsrichtlinie für Schreiboperationen 431
Platte
 entfernen 423
 hinzufügen 378
Platten (Festplattenlaufwerke) 386
 konfigurieren 386
Platten-Striping 439
Plattenbelegung
 Auswirkungen der Fragmente 490
Plattenbelegungsabrechnung 186
Platteninterne Zuordnung, Strategie 437

- Plattenlaufwerke
 - Dateisysteme auf einer Platte abhängen 466
 - diagnostizieren 396
 - Probleme beheben 396
 - siehe auch physische Datenträger 389
 - Speicherplatz freigeben 397
 - Speicherplatz von einer anderen Platte anhängen 398
 - veraltete Dateien entfernen 397
 - Wiederherstellung von Daten
 - ohne erneute Formatierung 398
 - Zugriff auf Verzeichnisse einschränken 397
- Plattenlaufwerke (Festplattenlaufwerke)
 - Ausfall
 - Wiederherstellung 403
 - Dateisysteme auflisten 466
- Plattenübergreifende Zuordnung, Strategie 434
- Plattenüberlauf beheben 478
- Platzhalterzeichen 199
 - Definition 219
 - Fragezeichen 200
 - Stern 200
- popd, integrierter Befehl
 - C-Shell 300
- Positionscodes 565
 - Adapter 565
 - definierte 565
 - Drucker/Plotter 566
 - Multiprotokollanschluss 568
 - nicht grafikfähiges Terminal 566
 - SCSI-Einheit 567
 - Wertgeber/Beleuchtete Funktionstasten 567
- Positionscodes für Adapter 565
- Positionsgebundene Parameter 239
 - Bourne-Shell 274
- PostScript-Dateien
 - von ASCII konvertieren 647, 648
- PostScript-Drucker
 - ASCII-Dateien drucken 647
- PPID, Variable 224
- pr, Befehl
 - Flags 646
- Primfaktoren, Zahlen zerlegen in
 - factor, Befehl 144
- print, integrierter Befehl
 - Korn- bzw. POSIX-Shell 253
- printenv, Befehl 332
- Priorität von Prozessen 160
- Priorität zuweisen
 - Druckjobs 641
- processMon 680, 681
- Profildateien 333
- profile
 - Dateien 55
 - Übersicht 55
- Programm
 - Ausgabe in eine Datei kopieren 367
- Protokoll
 - bearbeiten 140
 - Substitution in der C-Shell 308
- Prozesse 133
 - Abrechnungsberichte generieren 185
 - Abrechnungsdaten erfassen 185
 - alle aktiven anzeigen 147
 - an einen Prozessor binden 161
 - Anfangspriorität festlegen 148
 - aus dem Zeitplan entfernen 152
 - Beendigung 160

- Prozesse (Forts.)
 - Beschreibung 145
 - CPU-Belastung anzeigen 181
 - Dämon 145
 - für die Ausführung zu einem späteren Zeitpunkt planen 150
 - Geister- 145
 - geplante auflisten 151
 - gestoppte erneut starten 149
 - Hintergrund 145
 - Hintergrundprozesse entfernen 152
 - Priorität ändern 148, 160
 - Stapel 150
 - starten 146
 - Status überprüfen 147
 - überwachen 157
 - Verwaltung 157
 - Vordergrund 145
 - Vordergrundprozesse abbrechen 149
 - Vordergrundprozesse stoppen 149
- Prozessidentifikationsnummer 145
- Prozessübersichten 153
- Prüfen
 - Integrität von Bändern 52
- ps, Befehl 11, 147, 162, 253
- PS1, Variable 225
- PS2, Variable 225
- PS3, Variable 225
- PS4, Variable 225
- psh, Befehl 218, 267
- pushd, integrierter Befehl
 - C-Shell 300
- pwd, Befehl 509
- pwd, integrierter Befehl
 - Bourne-Shell 280
 - Korn- bzw. POSIX-Shell 253
- PWD, Variable 224

Q

- qcan, Befehl 640
- qchk, Befehl 645
- qmov, Befehl 642
- qpri, Befehl 641
- qpri, Befehl 637
 - Flags 637, 647
- Quorum
 - Datenträgergruppen ohne Quorum 374
 - Definition 372
 - in eine Datenträgergruppe ohne Quorum ändern 375

R

- r, Aliasname 139
- r, Befehl 139
- RANDOM, Variable 224
- Range-Einstellung 434
- read, integrierter Befehl
 - Bourne-Shell 280, 283
 - Korn- bzw. POSIX-Shell 253
- readonly, integrierter Befehl 246
 - Bourne-Shell 280
 - Korn- bzw. POSIX-Shell 247
- refresh, Befehl 196
- Reguläre Ausdrücke 201

- Reguläre integrierte Befehle
 - Korn- bzw. POSIX-Shell 230, 253
- rehash, integrierter Befehl
 - C-Shell 300
- Relativer Pfadname 508
- renice, Befehl 148, 162
- repDiskState 691
- repeat, integrierter Befehl
 - C-Shell 300
- REPLY, Variable 224
- Reservierte Wörter
 - Bourne-Shell 278
 - Korn- bzw. POSIX-Shell 234
- Ressourcendateien
 - ändern 337, 338
- Ressourcengruppen für exklusive Benutzung
 - Ressourcengruppe mit Prozessoren für exklusive Benutzung (XRSET) 551
- restore, Befehl 29, 53, 54
- Restricted Korn Shell
 - starten 234
- Restricted Shell 218
 - starten 223
- return, integrierter Befehl
 - Bourne-Shell 280
 - Korn- bzw. POSIX-Shell 247
- rm, Befehl 202, 214
- rmdir, Befehl 513
- rsh, Befehl 218, 223, 234
- runacct, Befehl
 - erneut starten 174
 - starten 173

S

- Schalter
 - in Befehlen 134
- schedo 679, 681
- Schlüsselwortsuche
 - apropos, Befehl 145
- Schreiben in die Monitordatei
 - Definition 659
- Schriftarten
 - ändern 340
 - verfügbare auflisten 329
- script, Befehl 368
- SCSI-Einheiten
 - Positionscodes 567
- SECONDS, Variable 224
- set, integrierter Befehl 274
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 247
- setclock, Befehl 67
- setenv, integrierter Befehl
 - C-Shell 300
- setgroups, integrierter Befehl
 - Korn- bzw. POSIX-Shell 253
- setsenv, integrierter Befehl
 - Korn- bzw. POSIX-Shell 253
- sh, Befehl 218
- Shared Product Object Tree (SPOT), Verzeichnis 462
- SHELL, Variable 225
- Shell-Scripts 133
 - erstellen 224
 - Shell angeben 221

- Shellbefehle
 - fc 140
 - history 137
 - r, Aliasname 139
- Shellprozeduren 133
- Shells
 - Aliassubstitution in der C-Shell 289
 - Anführungszeichen in Korn bzw. POSIX 232
 - Anmeldung 218
 - Bedingungsausdrücke in Korn bzw. POSIX 231
 - Befehle in Korn- bzw. POSIX-Shell verwenden 264
 - Befehlsausführung in der C-Shell 307
 - Befehlssubstitution in der C-Shell 306
 - Bourne 218
 - Bourne, bedingte Substitution 273
 - Bourne, Befehlssubstitution 283
 - Bourne, benutzerdefinierte Variablen 284
 - Bourne, Dateinamenssubstitution 274
 - Bourne, E/A-Umleitung 275
 - Bourne, integrierte Befehle 279
 - Bourne, Liste der integrierten Befehle 276
 - Bourne, positionsgebundene Parameter 274
 - Bourne, Umgebung 272
 - Bourne, Variablensubstitution 284
 - Bourne, verwendete Variablen 285
 - Bourne, vordefinierte Variablen 287
 - Bourne, Zeichenklassen 222
 - Bourne-Shell starten 271
- C 218
- C, integrierte Befehle 299, 300
- C-Shell, Eingabe und Ausgabe umleiten 310
- C-Shell starten 288
- Dateinamenssubstitution in der C-Shell 292
- fern 218
- Funktionen 221
- Jobsteuerung in der C-Shell 296
- Korn 218
- Korn- bzw. POSIX-Shell, reservierte Wörter 234
- Korn bzw. POSIX, arithmetische Berechnungen 228
- Korn bzw. POSIX, Befehls-Aliasing 270
- Korn bzw. POSIX, Befehlsprotokoll 269
- Korn bzw. POSIX, Befehlssubstitution 227
- Korn bzw. POSIX, Befehlszeileneditierung 257
- Korn bzw. POSIX, Dateinamenssubstitution 242
- Korn bzw. POSIX, E/A-Umleitung 243
- Korn bzw. POSIX, Exit-Status 238
- Korn bzw. POSIX, integrierte Befehle 246
- Korn bzw. POSIX, Jobsteuerung 256
- Korn bzw. POSIX, Koprozesse 245
- Korn bzw. POSIX, Liste der integrierten Sonderbefehle 230
- Korn bzw. POSIX, Liste der regulären integrierten Befehle 230
- Korn bzw. POSIX, Signalverarbeitung 257
- Korn bzw. POSIX, starten 267
- Korn bzw. POSIX, Umgebung 267
- Korn bzw. POSIX, zusammengesetzte Befehle 265
- Parameter 239
- POSIX 218
- Protokolllisten in der C-Shell 308
- Protokollsubstitution in der C-Shell 308
- Restricted 218
- Restricted, starten 223, 234
- Shell-Scripts erstellen 224
- Signalverarbeitung in der C-Shell 299
- Standard 218
- Terminologie 219

- Shells (*Forts.*)
 - Trusted 218
 - Typen 218
 - Übersicht 217
 - Umgebungsvariablen in der C-Shell 294
 - Variablensubstitution in der C-Shell 290
- Shellumgebungen
 - anpassen 55
- Shellvariablen
 - Definition 219
 - exportieren 339
 - lokal 339
- shift, integrierter Befehl 274
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 247
- shutdown, Befehl 136
- Sicherheit
 - Anmelde-ID 312
 - Authentifizierung 312
 - Datei 312
 - Datei /etc/security/passwd 312
 - Identifikation 312
 - System 312
 - unbeaufsichtigte Terminals 313
- Sichern 43, 44
 - Befehle, Liste 22
 - Benutzerdateien 28
 - Benutzerdateisysteme 28
 - benutzerdefinierte Datenträgergruppen 44
 - BSD-Systemmanager 348
 - Dateien 22
 - Dateien komprimieren 40
 - Dateien zuvor komprimieren 40
 - Daten wiederherstellen 29
 - Methoden 22
 - mit dem Befehl smit 54
 - Prozedur für Benutzerdateisysteme 28
 - Prozedur für System- und Benutzerdaten 27
 - Richtlinie 24
 - Übersicht 22
 - Verwaltungsstrategie
 - planen 26
 - Richtlinien 24
- Sicherung
 - Auswirkung von Fragmenten 503
 - Dateien wiederherstellen 33
 - Datenträger 25
 - mit Scripts implementieren 47
 - regelmäßige geplante durchführen 47
 - System replizieren (klonen) 28
- SIGINT, Signal 257
- Signale
 - SIGINT 257
 - SIGQUIT 257
- Signalverarbeitung
 - Bourne-Shell 278
 - C-Shell 299
 - Korn- bzw. POSIX-Shell 257
- SIGQUIT, Signal 257
- skulker, Befehl 397
- smit, Befehl 54, 340
 - ASCII in PostScript konvertieren 648
 - Dateien wiederherstellen 29
 - Druckjob abbrechen 641
 - Druckjob starten 637
 - Druckjob verschieben 642
- smit, Befehl (*Forts.*)
 - Druckjobs Prioritäten zuweisen 641
 - Status eines Druckjobs prüfen 645
- smit rmat, Befehl 152
- Software
 - auf Einheitenfehler überprüfen 595
- Softwareprodukte
 - Informationen anzeigen 330
- Sonderparameter 239
- sort, Befehl 207
- Sortieren
 - Textdateien 207
- source, integrierter Befehl
 - C-Shell 300
- Speichermedien 22
- Speicherplatz
 - verfügbar 464
- Speicherung logischer Datenträger
 - Definition 414
 - logische Datenträger 417
 - logische Partitionen 417
 - maximale Größen 418
 - Plattenüberlauf 478
 - Richtlinie für platteninterne Zuordnung 437
 - Richtlinie für plattenübergreifende Zuordnung 434
- Sperren
 - Terminal 326
- Spiegelkonsistenz (MWC) 432
- Spiegeln
 - Datenträgergruppe 419
 - einen gespiegelten Datenträger von einer Datenträgergruppe abtrennen 395
 - Stammdatenträgergruppe (rootvg) 420
- Spiegelung entfernen
 - Datenträgergruppe 422
- SPOT-Verzeichnis 462
- Sprachen
 - bidirektional 332
- srcmstr, Befehl 196
- srcmstr, Dämon 194
- Stammdatensystem 455
- Stammdatensystem (/) 457
- Stammdatenträgergruppe (rootvg)
 - Größe von Dateisystemen verringern 472
 - spiegeln 420
- Standardausgabe 363
 - an eine Datei anfügen 364
 - umleiten 363
- Standardeingabe 363
 - in eine Datei kopieren 367
 - umleiten 365
- Standardfehler 363
- Standardfehlerausgabe
 - umleiten 365
- Standardshell 218
 - Bedingungsausdrücke 231
- Standardwerte
 - ändern 337
- Stapelprozesse 150
- Startdateien
 - AIXwindows 336
 - C-Shell 288
 - System 333
 - X-Server 336
- Starten
 - AIXwindows-Fenstermanager 336
 - Bourne-Shell 271

Starten (*Forts.*)
 C-Shell 288
 Druckjobs 637
 Korn- bzw. POSIX-Shell 267
 Prozesse 146
 Restricted Korn Shell 234
 Restricted Shell 223
 Starten von Workload Manager 523
 startsrc, Befehl 195
 Status
 Drucker 644, 645
 Statusbereich der Datenträgergruppe (VGSA) 371
 stderr 363
 stdin 363
 stdout 363
 Steuermodus 260
 Steuertasten
 ändern 340
 Einstellungen anzeigen 330
 stop, integrierter Befehl
 C-Shell 300
 Stoppen
 Vordergrundprozesse 149
 Stoppen von Workload Manager 523
 stopsrc, Befehl 195
 strict, Einstellung für plattenübergreifende Zuordnung 436
 Strl-C, Tastenkombination 11
 stty, Befehl 330, 340
 su, Befehl 322, 626
 Subserver
 Beschreibung 193
 starten 195
 Status anzeigen 196
 stoppen 195
 Traceerstellung aktivieren 197
 Traceerstellung inaktivieren 197
 Subshells
 Definition 219
 Subsystem
 aktualisieren 196
 Eigenschaften 193
 starten 195
 Status anzeigen 196
 stoppen 195
 Traceerstellung aktivieren 197
 Traceerstellung inaktivieren 197
 Subsystemgruppe
 aktualisieren 196
 Beschreibung 193
 starten 195
 Status anzeigen 196
 stoppen 195
 Traceerstellung aktivieren 197
 Traceerstellung inaktivieren 197
 suchen
 Schlüsselwörter 145
 Suchen
 Dateien 204
 Textzeichenfolgen in Dateien 207
 super strict, Einstellung für plattenübergreifende Zuordnung 436
 suspend, integrierter Befehl
 C-Shell 300
 switch, integrierter Befehl
 C-Shell 300
 System
 Abrechnung 624
 System (*Forts.*)
 Eingabeaufforderung ändern 341
 einschalten 625
 Namen anzeigen 629
 Sicherheit 312
 Standardvariablen 334
 Startdateien 333
 starten 4
 Umgebung 326
 Umgebung anpassen 339, 340, 341
 Verwaltung 453
 System Resource Controller
 Befehle
 Liste 194
 Funktionen 192
 starten 194
 Systemabrechnung
 Befehl runacct
 erneut starten 174
 starten 173
 Befehle
 automatische Ausführung 170
 über die Tastatur ausführen 171
 Berichte
 Finanz- 169
 Monats- 167, 168
 Tages- 165
 Berichtsdaten
 Übersicht 165
 CPU-Belastung
 anzeigen 181
 Dateien
 Berichts- und Übersichtsdateien 173
 Dateien des Befehls runacct 174
 Datendateien 172
 Formate 176
 Übersicht 172
 Daten zur Druckerauslastung 182
 berichten 169
 erfassen 186
 Daten zur Plattenbelegung 168, 182
 erfassen 186
 Daten zur Systemaktivität
 anzeigen 179
 während der Ausführung eines Befehls anzeigen 179
 Daten zur Verbindungsdauer 168, 182, 184
 Datensätze zusammenfassen 167
 Fehler
 Wiederherstellung 174
 Gebühren
 berechnen 187
 berichten 169
 holidays, Datei
 aktualisieren 183
 konfigurieren 176
 Probleme
 nicht mehr auf dem neuesten Stand befindliche Datei
 holidays korrigieren 183
 runacct-Fehler beheben 190
 ungültige Dateiberechtigungen korrigieren 189
 ungültige Zeitangaben beheben 189
 Prozessdaten
 berichten 185
 erfassen 185
 Systemaktivität
 Daten 169

- Systemabrechnung (*Forts.*)
 - tacct-Fehler
 - beheben 187
 - Übersicht 164
 - wtmp-Fehler
 - beheben 188
- Systemabschluss 55
 - in den Einzelbenutzermodus 55
 - Informationen 54
 - Notfall 55
 - ohne Warmstart 55
- Systemabschluss durchführen 136
- Systemaktivität
 - überwachen 169
- Systemausfall
 - Hardware überprüfen 10
 - Prozesse überprüfen 11
 - System erneut starten 13
- Systemausführungsebene 16
- Systembatterie 65
- Systemstart
 - Fenster und Anwendungen steuern 336
- Systemuhr
 - Batterie testen 65
 - zurücksetzen 66
- Systemumgebung 58
 - 64-Bit-Modus 57
 - dynamische Prozessorfreigabe 58, 59
 - Nachricht des Tages 67
 - profile 55
 - Services für die Bearbeitung von Zeitdaten 56

T

- Tabelle
 - Tastatur 330
- tacct-Fehler
 - beheben 187
- tail, Befehl 209
- tapechk, Befehl 24, 52
- tar, Befehl 25, 40, 52
- Tastatur
 - Attribute ändern
 - Befehl chhwkbd verwenden 357
- Tastaturtabellen
 - derzeit verfügbare auflisten 330
- Tastenbelegungen 338
- tcopy, Befehl 52
- TCP/IP
 - /etc/hosts 19
 - Benennung
 - einfaches Netz 19
 - hierarchisches Netz 19
 - Netzplanung 19
- tee, Befehl 367
- Terminal, gesperrt 162
- Terminalprobleme
 - blockierte Prozesse stoppen 162
- Terminals
 - Einstellungen anzeigen 332
 - für BSD-Systemmanager 362
 - Namen anzeigen 329
 - sperrern 326
 - Steuertastenkombinationen anzeigen 330
 - unbeaufsichtigt 313
- Terminologie
 - für Shells 219

- test, integrierter Befehl
 - Bourne-Shell 280
 - Korn- bzw. POSIX-Shell 253
- Text
 - an eine Datei anfügen 368
 - in großen Buchstaben anzeigen 369
- Textdateien
 - Abschnitte ausschneiden 210
 - Abschnitte einfügen 210
 - aus Tastatureingabe erstellen 364
 - sortieren 207
 - Spalten entfernen 212
 - verknüpfen 364
 - Zeichenfolgen suchen 207
 - Zeilen nummerieren 211
- Textformatierung
 - Befehle 141
 - erweiterte Einzelbytezeichen 142
 - Unterstützung internationaler Zeichensätze 141
 - Unterstützung von Mehrbytezeichen 142
- Textformatierungssystem 647
- Textzeile
 - in einer Datei anfügen 368
- Tildensubstitution
 - Befehls-Aliasing 271
- time, integrierter Befehl
 - C-Shell 300
- times, integrierter Befehl
 - Bourne-Shell 280
 - Korn- bzw. POSIX-Shell 247
- TMOUT, Variable 225
- tn3270, Befehl 342
- touch, Befehl 626
- tracesoff, Befehl 197
- traceson, Befehl 197
- Transmission Control Protocol/Internet Protocol 19
- trap, integrierter Befehl
 - Bourne-Shell 280
 - Korn- bzw. POSIX-Shell 247
- Trusted Shell 218
- tsh, Befehl 218
- tty, Befehl 329
- type, integrierter Befehl
 - Bourne-Shell 280
- typeset, integrierter Befehl 246
 - Korn- bzw. POSIX-Shell 228, 239, 247, 267

U

- Übergeordnetes Verzeichnis 507
- Überprüfen
 - Prozessstatus 147
 - Status von Druckjobs 645
- Überprüfung von Dateisystemen 470
- Überprüfung von Dateisystemen auf Inkonsistenzen 470
- Übersichten
 - Befehle 369
 - für Befehle 153
 - für Drucken 649
 - für Prozesse 153
 - Startdateien für AIXwindows 327
 - Systemstartdateien 327
 - Systemumgebung anpassen 327
- Überwachung von Prozessen 157
- Uhr
 - zurücksetzen 66

- ulimit, integrierter Befehl
 - Bourne-Shell 280
 - Korn- bzw. POSIX-Shell 253
- umask, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 253
- Umbenennen
 - Dateien 202
 - Verzeichnisse 509
- Umgebung
 - aktuelle anzeigen 331
 - Datei 333
 - Einstellung 333
 - System 326
- Umgebungsvariablen
 - C-Shell 294
 - Werte anzeigen 332
- Umleiten
 - Ausgabe in eine Datei 364
 - Eingabe und Ausgabe in der Bourne-Shell 275
 - Eingabe und Ausgabe in der Korn- bzw. POSIX-Shell 243
 - Eingabe und Ausgabe von Koprozessen 246
 - Standardausgabe 363
 - Standardeingabe 365
 - Standardfehlerausgabe 365
- unalias, integrierter Befehl
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 253, 270
- uname, Befehl 628, 629
- uncompress, Befehl 40, 41, 42
- unhash, integrierter Befehl
 - C-Shell 300
- unlimit, integrierter Befehl
 - C-Shell 300
- unpack, Befehl 40, 42
- unset, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 247
- unsetenv, integrierter Befehl 300
- Unterstreichungszeichen, Variable 224
- Unterstützung internationaler Zeichensätze
 - Textformatierung 141
- Unterstützung von Mehrbytezeichen
 - Textformatierung 142
 - Zeichen eingeben 142
- USB-Blu-ray-Laufwerk, Unterstützung 614
- USB-Einheit, Unterstützung 613
- USB-Flashlaufwerk 614
- utilFs 675, 681
- UUCP
 - BSD-Systemmanager 360

V

- Variable Anzahl von I-Nodes 492
 - und Fragmente 490
- Variablen
 - benutzerdefinierte 225
 - benutzerdefinierte in der Bourne-Shell 284
 - Bourne-Shell 285, 287
 - C-Shellumgebung 294
 - CDPATH 225
 - COLUMNS 225
 - EDITOR 225
 - ENV 225

- Variablen (*Forts.*)
 - ERRNO 224
 - exportieren 339
 - FCEDIT 225
 - FPATH 225
 - HISTFILE 225
 - HISTSIZE 225, 269
 - HOME 225
 - IFS 225
 - Korn- bzw. POSIX-Shell 224, 225
 - LANG 225
 - LC_ALL 225
 - LC_COLLATE 225
 - LC_CTYPE 225
 - LC_MESSAGES 225
 - LINENO 224
 - LINES 225
 - MAIL 225
 - MAILCHECK 225
 - MAILPATH 225
 - NLSPATH 225
 - OLDPWD 224
 - OPTARG 224
 - OPTIND 224
 - PATH 225
 - PPID 224
 - PS1 225
 - PS2 225
 - PS3 225
 - PS4 225
 - PWD 224
 - RANDOM 224
 - REPLY 224
 - SECONDS 224
 - SHELL 225
 - SHELL PROMPT, Variable 225
 - TMOUT 225
 - Unterstreichungszeichen 224
 - Variablen
 - SHELL PROMPT 225
 - VISUAL 225
 - vordefinierte 224
- Variablensubstitution
 - Bourne-Shell 284
 - C-Shell 290
 - Korn- bzw. POSIX-Shell 224
- Verbinden
 - Dateien 213
- Verbindung
 - Dateien 212
 - Verzeichnisse 212
- Verbindungen
 - erstellen 213
 - feste 213
 - symbolische 213
 - Typen 213
 - Übersicht 212
- Verfügbarkeit
 - bei Adapter- oder Netzteilausfall 429
 - bei Plattenausfall 429
- Vergleichen von Dateien 208
- Verketteten 133
- Verknüpfen
 - Textdateien 364
- Verknüpfte Dateien
 - entfernen 214

- Verknüpfungen
 - entfernen 214
- Verlagern
 - Adapter für DLPAR 652
- Verschieben
 - Druckjobs 642
- Verwalten
 - ACLs 318
 - Zugriffssteuerungslisten 318
- Verwaltung 376
- Verwendungsanweisungen
 - für Befehle 135
- Verwerfen der Ausgabe 365
- Verzeichnisse 505
 - Abkürzungen 507
 - anhängen 483
 - anzeigen 509
 - Arbeits- 507
 - Ausgangs- 507
 - Berechtigungen 314
 - Berechtigungen ändern 317
 - Dateien auflisten 511
 - DOS-Dateien auflisten 216
 - Eigentumsrecht ändern 313
 - entfernen 513
 - erstellen 508
 - Inhalt anzeigen 511
 - Inhalt vergleichen 513
 - kopieren 510
 - löschen 513
 - mit Abkürzungen angeben 507
 - Namenskonventionen 507
 - Organisation 507
 - Pfadnamen 508
 - Stamm- 505
 - Struktur 507
 - Typen 506
 - übergeordnete 507
 - Übersicht 505
 - umbenennen 509
 - Unterverzeichnisse 507
 - Verbindung 212
 - verschieben 509
 - wechseln 510
 - Zugriffsmodi 314
- Verzeichnisstruktur 507
- VGDA (Volume Group Descriptor Area) 371
- VGSA (Volume Group Status Area) 371
- vgState 692
- vi, Editor 201
 - Befehle für Eingabeeditierung 261
 - Befehlszeileneditierung 257, 260, 261, 262, 263
 - Cursorbewegung 261
 - Editierbefehle für Cursorbewegung 261
 - Editierbefehle für Textänderung 262
 - Eingabemodus 260, 261
 - häufig verwendete Editierbefehle 263
 - Steuermodus 260
 - Sucheditierbefehle 262
 - verschiedene Editierbefehle 263
- Virtual Memory Manager 451
- Virtual Memory Manager (VMM)
 - Übersicht 442
- VISUAL, Variable 225
- VMM 451
- vmo 677, 681

- vordefinierte Variablen
 - Korn- bzw. POSIX-Shell 224
- Vordefinierte Variablen
 - Bourne-Shell 287
- Vordergrundprozesse 145
- Vorkommen von Nichtverfügbarkeitsereignissen
 - Definition 663

W

- wait, integrierter Befehl
 - Bourne-Shell 280
 - C-Shell 300
 - Korn- bzw. POSIX-Shell 253
- waitersFreePg 676, 681
- waitTmCPU 676, 681
- waitTmPgInOut 677, 681
- Warmstart eines Systems mit einem integrierten Grafiksysteem 8
- Warten auf Ereignisse
 - Definition 662
- Warteschlange
 - Status 645
- wc, Befehl 208
- Wechseln
 - in ein anderes Verzeichnis 510
- Wertgeber/Beleuchtete Funktionstasten, Positionscodes 567
- whatis, Befehl 137
- whence, integrierter Befehl
 - Korn- bzw. POSIX-Shell 253
- whereis, Befehl 136
- while, integrierter Befehl
 - C-Shell 300
- who, Befehl 162, 627, 629
- who am i, Befehl 627
- whoami, Befehl 627
- Wiederherstellen
 - Dateien 29, 54
- Wiederherstellung
 - Auswirkung von Fragmenten 503
- Wiederherstellungsprozeduren
 - Warmstart eines Systems mit einem integrierten Grafiksysteem 8
 - Zugriff auf ein System, das nicht bootet 21
- Wiederherstellungsprozeduren für den Ausfall von Plattenlaufwerken
 - Beispiel 403
- WLM
 - API 558
- Workload Manager
 - API 558
 - starten und stoppen 523
- Workstations ohne Plattenspeicher
 - Mountsicherheit 485
- Wörter
 - Definition 219
 - reserviert in Korn- bzw. POSIX-Shell 234
 - zählen 208
- write-verify-Richtlinie 439
- wtmp-Fehler
 - beheben 188

X

- X-Server
 - Startdateien 336

X-Terminal 633
xinit, Befehl 336
xlock, Befehl 326

Y

Yellow Pages 347
 BSD-Systemmanager 347

Z

Zählen
 Bytes 208
 Wörter 208
 Zeilen 208
zcat, Befehl 42
Zeichen
 Anführungszeichen in Korn- bzw. POSIX-Shell 232
Zeichen in Anführungszeichen setzen
 Bourne-Shell 277
 Korn- bzw. POSIX-Shell 232
Zeichenfolgen
 in Textdateien suchen 207
Zeichenklassen
 Bourne-Shell 222
Zeichenorientierte Terminals
 hinzufügen 633
Zeilen
 zählen 208
Zeitmanagement
 calendar, Befehl 143
 Erinnerungsnachrichten 143
 schreiben, Erinnerungsnachrichten 143
Zieleinheitenkonfiguration 599
Zugeordneten Speicherbereich für Dateien mit Nullen auffüllen (kproc) 496
Zugriff auf ein System, das nicht bootet 21
Zugriff der Benutzer auf bestimmte Verzeichnisse einschränken 397
Zugriffsmodi
 Benutzerklassen 314
 Dateien 314
 festlegen 314
 Gruppeninformationen anzeigen 316
 numerische Darstellung 315
 symbolische Darstellung 314
 Verzeichnisse 314
Zugriffssteuerung
 Informationen anzeigen 324
 Informationen editieren 325
 Informationen festlegen 324
Zugriffssteuerungslisten 318
 ACL-Beispiel für AIXC 321
 Beispiel 324
 für Dateisystemobjekte 318
 verwalten 318
Zuordnen
 Werte und Attribute 239
Zuordnungsdateien 438
Zuordnungsgruppe, Größe 494
Zusammengesetzte Befehle 265
 Bourne-Shell 278



Gedruckt in Deutschland