

IBM Communications Server für Linux



# Verwaltungshandbuch

*Version 6.2.3*



IBM Communications Server für Linux



# Verwaltungshandbuch

*Version 6.2.3*

**Hinweis**

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten unbedingt die allgemeinen Informationen in Anhang C, „Bemerkungen“, auf Seite 199 gelesen werden.

**Vierte Ausgabe (Dezember 2007)**

Diese Ausgabe bezieht sich auf IBM Communications Server für Linux Version 6.2.3 und, sofern in neuen Ausgaben oder TNLs nichts anderes angegeben ist, auf alle folgenden Releases und Änderungen.

Veröffentlichungen können über den zuständigen IBM Ansprechpartner oder die zuständige IBM Geschäftsstelle bezogen werden. Veröffentlichungen sind nicht bei der unten angegebenen Adresse erhältlich.

Am Ende des Handbuchs finden Sie einen Vordruck für einen Antwortbrief.

Werden an IBM Informationen eingesandt, gewährt der Einsender IBM ein nicht ausschließliches Recht zur beliebigen Verwendung oder Verteilung dieser Informationen, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs  
*IBM Communications Server for Linux Administration Guide, Version 6.2.3*,  
IBM Form SC31-6771-02,  
herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 1998, 2007

© Copyright IBM Deutschland GmbH 2007

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:  
SW TSC Germany  
Kst. 2877  
Oktober 2007

# Inhaltsverzeichnis

<b>Tabellen</b> . . . . .	<b>vii</b>
---------------------------	------------

<b>Abbildungsverzeichnis</b> . . . . .	<b>ix</b>
--	-----------

<b>Zu diesem Handbuch.</b> . . . . .	<b>xi</b>
--------------------------------------	-----------

Zielgruppe . . . . .	xi
Benutzung des Handbuchs . . . . .	xii
Aufbau des Handbuchs . . . . .	xii
Typografische Konventionen . . . . .	xii
Konventionen der grafischen Darstellung . . . . .	xiii
Neuerungen in diesem Release . . . . .	xiv
Neue Funktionen . . . . .	xiv
Nicht mehr unterstützte Funktionen . . . . .	xv
Referenzliteratur. . . . .	xv

## **Kapitel 1. SNA-Begriffe und -Konzepte . 1**

Systemnetzwerkarchitektur . . . . .	1
SNA-Basiskonzepte . . . . .	2
Netztypen . . . . .	2
SNA-Knoten . . . . .	2
Konnektivität . . . . .	5
Transaktionsprogramme . . . . .	5
Anwendungsprogrammierschnittstellen . . . . .	6
Adressierbare Netzeinheiten . . . . .	6
Sitzungen . . . . .	9
Dialoge . . . . .	11
Modi . . . . .	12
Routenauswahl . . . . .	13
Serviceklasse . . . . .	13
APPN-Basiskonzepte . . . . .	14
Arten von APPN-Knoten . . . . .	14
APPN-Steuerpunkt . . . . .	16
Ressourcen lokalisieren . . . . .	17
Sitzungs-Routing . . . . .	21
Branch Extender . . . . .	28
Von APPN-Netzen auf Unterbereichsnetze zugreifen . . . . .	30

## **Kapitel 2. Communications Server für Linux verwalten . . . . . 31**

Verwaltung von Communications Server für Linux im Überblick . . . . .	31
Verwaltungszuständigkeit . . . . .	31
Verwaltungstools . . . . .	32
Berechtigungen für die Verwaltung . . . . .	38
CS-Linux-Konfiguration planen. . . . .	38
Planungsblätter . . . . .	39
Aufgabenblätter . . . . .	39
Communications Server für Linux auf dem lokalen System aktivieren und inaktivieren . . . . .	40
Pfad zu CS-Linux-Programmen angeben. . . . .	40
CS-Linux-Server aktivieren . . . . .	40
CS-Linux-Server inaktivieren . . . . .	42
Motif-Verwaltungsprogramm verwenden . . . . .	43
Motif-Verwaltungsprogramm aufrufen . . . . .	43

Ressourcenfenster . . . . .	44
Dialoganzeigen für Ressourcen . . . . .	51
Statusdialoganzeigen . . . . .	53
Hilfefenster . . . . .	54
Befehlszeilenverwaltungsprogramm verwenden . . . . .	55

## **Kapitel 3. Grundlegende Konfigurationsaufgaben . . . . . 57**

Client-Server-Funktionen konfigurieren . . . . .	57
Knoten konfigurieren . . . . .	58
Parameter für die Knotenkonfiguration . . . . .	59
Zusätzliche Konfiguration . . . . .	59
Protokollierung konfigurieren . . . . .	60

## **Kapitel 4. Konnektivitätskomponenten definieren . . . . . 63**

DLCs, Ports und Verbindungsnetze definieren. . . . .	64
Konfigurationsparameter für DLCs, Verbindungsnetze und Ports . . . . .	65
Zusätzliche Konfiguration . . . . .	69
Verbindungsstationen definieren . . . . .	69
Konfigurationsparameter für Verbindungsstationen . . . . .	70
Zusätzliche Konfiguration . . . . .	77
DLUR-PUs definieren . . . . .	77
Konfigurationsparameter für DLUR-PU. . . . .	78
Parameter für Durchgriff-DLUR für untergeordnete Knoten . . . . .	79
Zusätzliche Konfiguration . . . . .	80

## **Kapitel 5. Abhängige LUs konfigurieren 81**

LUs der Typen 0–3 definieren . . . . .	81
Konfigurationsparameter für LUs der Typen 0–3 . . . . .	82
Zusätzliche Konfiguration . . . . .	83
LU-Pools definieren . . . . .	83
Konfigurationsparameter für LU-Pools . . . . .	84

## **Kapitel 6. APPC-Kommunikation konfigurieren . . . . . 85**

Lokale LUs definieren . . . . .	86
Konfigurationsparameter für lokale LUs. . . . .	87
Zusätzliche Konfiguration . . . . .	87
Ferne Knoten definieren . . . . .	88
Konfigurationsparameter für ferne Knoten . . . . .	89
Zusätzliche Konfiguration . . . . .	89
Partner-LUs definieren. . . . .	89
Konfigurationsparameter für Partner-LUs . . . . .	90
Routenwahl für Verbindungsstationen für eine Partner-LU definieren . . . . .	92
Zusätzliche Konfiguration . . . . .	93
TPs definieren . . . . .	93
Aufrufparameter für Transaktionsprogramme auf einem Server . . . . .	96
Parameter für TP-Definition . . . . .	99

Modi und Serviceklassen definieren . . . . .	100
Konfigurationsparameter für Modi . . . . .	102
Zusätzliche Konfiguration . . . . .	105
CPI-C-Nebeninformationen definieren . . . . .	105
Konfigurationsparameter für CPI-C . . . . .	106
Zusätzliche Konfiguration . . . . .	108
APPC-Sicherheit konfigurieren . . . . .	108
Sitzungssicherheit konfigurieren . . . . .	108
Dialogsicherheit konfigurieren . . . . .	109
Sicherheitszugriffsliste konfigurieren . . . . .	110

## **Kapitel 7. Benutzeranwendungen konfigurieren . . . . . 111**

## **Kapitel 8. Durchgriffsdienste konfigurieren . . . . . 113**

TN Server konfigurieren . . . . .	113
TN-Server-Zugriffseinträge konfigurieren . . . . .	114
TN-Server-Zuordnungseinträge konfigurieren . . . . .	117
TN Redirector konfigurieren . . . . .	118
TN-Redirector-Zugriffseinträge konfigurieren . . . . .	118
SNA Gateway konfigurieren . . . . .	122
Konfigurationsparameter für untergeordnete LUs . . . . .	124
Zusätzliche Konfiguration . . . . .	124
DLUR konfigurieren . . . . .	124

## **Kapitel 9. Communications Server für Linux mit NetView verwalten . . . . . 127**

Hostprogramm NetView verwenden . . . . .	127
NetView-Bildschirmanzeige . . . . .	128
Größe des Eingabebereichs ändern . . . . .	128
Übersicht über die Syntax der RCF-Befehle . . . . .	128
Großbuchstaben und Escape-Zeichen . . . . .	129
SPCF verwenden . . . . .	130
Einschränkungen für über SPCF abgesetzte Verwaltungsbefehle . . . . .	130
Beispiele für SPCF-Befehle . . . . .	131
UCF verwenden . . . . .	131
Syntax der UCF-Befehle . . . . .	132
Zulässige Befehle . . . . .	132
Beispiel für einen UCF-Befehl . . . . .	133
Ausgabe von Linux-Systembefehlen . . . . .	133
Befehl abbrechen . . . . .	134
UCF-Sicherheit . . . . .	134

## **Kapitel 10. Client-Server-Systeme mit Communications Server für Linux verwalten . . . . . 137**

Client-Server-Konfiguration ändern . . . . .	138
Clients in eine andere Domäne verschieben . . . . .	139
Voraussetzung für den IP-Netzbetrieb . . . . .	139
IPV4- und IPV6-Adressen . . . . .	139
Hostnamen in der Client-Server-Konfiguration . . . . .	140
IP-Port-Nummern konfigurieren . . . . .	140
Zeitlimit für LAN-Zugriff . . . . .	141
HTTPS-Zugriff für Remote API Clients . . . . .	142
Remote API Clients unter Windows verwalten . . . . .	143
Remote API Client unter Windows aktivieren . . . . .	144

Status eines Remote API Client unter Windows anzeigen . . . . .	144
Remote API Client unter Windows inaktivieren . . . . .	144
Konfiguration für den Remote API Client unter Windows . . . . .	145
Remote API Clients unter AIX oder Linux verwalten . . . . .	159
Remote API Clients unter AIX oder Linux aktivieren . . . . .	159
Netzdatendatei des Clients (sna_clnt.net) . . . . .	160
Client-TPs definieren . . . . .	164

## **Anhang A. Planungsblätter für die Konfiguration . . . . . 165**

Arbeitsblätter für Knoten . . . . .	165
APPN-Netzknoten . . . . .	165
APPN-Endknoten . . . . .	166
APPN-Zweignetzknoten . . . . .	167
LEN-Knoten . . . . .	167
Arbeitsblätter für Konnektivität . . . . .	168
SDLC . . . . .	168
Token-Ring . . . . .	170
Ethernet . . . . .	172
QLLC (X.25) . . . . .	174
MultiPath Channel . . . . .	175
Enterprise Extender (HPR/IP) . . . . .	176
Arbeitsblätter für Durchgriffsdienste . . . . .	177
DLUR auf dem lokalen Knoten . . . . .	178
Durchgriff-DLUR für untergeordnete Knoten . . . . .	178
SNA Gateway . . . . .	179
TN Server . . . . .	179
TN Redirector . . . . .	180
Arbeitsblätter für Anwendungsunterstützung . . . . .	181
APPC . . . . .	181
CPI-C . . . . .	185
5250 . . . . .	185
3270 . . . . .	186
LUA . . . . .	187

## **Anhang B. Aufrufbares TP mit Befehlszeilendienstprogramm konfigurieren . . . . . 189**

Format der Definitionsdatei für aufrufbares TP . . . . .	190
--	-----

## **Anhang C. Bemerkungen . . . . . 199**

Marken . . . . .	201
------------------	-----

## **Literaturverzeichnis . . . . . 203**

Veröffentlichungen zu Communications Server für Linux Version 6.2.3 . . . . .	203
Veröffentlichungen zur Systemnetzwerkarchitektur (SNA) . . . . .	204
Veröffentlichungen zur Hostkonfiguration . . . . .	205
Veröffentlichungen zu z/OS Communications Server . . . . .	205
Veröffentlichungen zu TCP/IP . . . . .	205
Veröffentlichungen zu X.25 . . . . .	205
Veröffentlichungen zu APPC . . . . .	205
Veröffentlichungen zur Programmierung . . . . .	206

Veröffentlichungen zu anderen IBM Produkten für  
den Netzbetrieb . . . . . 206

**Index . . . . . 207**



---

## Tabellen

1.	Typografische Konventionen . . . . .	xiii	3.	Verwendung von Escape-Zeichen in RCF-Befehlen . . . . .	130
2.	Namen der Standardmodi und Serviceklassen	101			



---

## Abbildungsverzeichnis

1.	SNA-Unterbereichsnetz . . . . .	4	11.	Für direkte Verbindungen über einen virtuel-	28
2.	Mehrfachsitzungen und Parallelsitzungen	11	len Knoten benötigte Definitionen . . . . .	28	
3.	Kommunikation zwischen Transaktions-	12	12.	Branch Extender . . . . .	29
	programmen und logischen Einheiten. . . . .	12	13.	Domänenfenster für Communications Server	
4.	Ausschnitt aus einem APPN-Beispielnetz	15	für Linux . . . . .	45	
5.	LEN-Knotenverzeichnis . . . . .	19	14.	Knotenfenster . . . . .	47
6.	Endknotenverzeichnis . . . . .	20	15.	Funktionsleiste für Communications Server für	
7.	Netzknotenverzeichnis . . . . .	20	Linux . . . . .	49	
8.	Netztopologiedatenbank auf Netzknoten	23	16.	Beispieldialoganzeige . . . . .	51
9.	APPN-Netz mit SATF-Einrichtung . . . . .	26	17.	Beispiel für eine Statusdialoganzeige . . . . .	53
10.	Für direkte Verbindungen zwischen dem Kno-	26	18.	Beispiel für ein Hilfefenster . . . . .	54
	ten EK1 und allen anderen Knoten eines		19.	Beispiel für eine NetView-Anzeige . . . . .	128
	APPN-Netztes erforderliche Definitionen . . . . .	26			



---

## Zu diesem Handbuch

Communications Server für Linux ist ein IBM Softwareprodukt, das einem Linux-Computer ermöglicht, Informationen mit anderen Knoten in einem SNA-Netz auszutauschen. Das vorliegende Handbuch beschäftigt sich mit der Aktivierung, Konfiguration und Verwaltung von IBM Communications Server für Linux.

Die Installationsvariante für IBM Communications Server für Linux richtet sich nach der Hardware, auf der das Produkt ausgeführt wird:

### **Communications Server für Linux**

Communications Server für Linux, Lizenzprogrammnummer 5724-i33, kann auf folgenden Systemen ausgeführt werden:

- 32-Bit Intel-Workstations mit Linux (i686)
- 64-Bit-AMD64/Intel-EM64T-Workstations mit Linux (x86\_64)
- IBM pSeries-Computer mit Linux (ppc64)

### **Communications Server für Linux auf System z**

Communications Server für Linux auf System z, Lizenzprogrammnummer 5724-i34, wird auf System-z-Großrechnern mit Linux für System z (s390 oder s390x) ausgeführt.

Im vorliegenden Handbuch wird der Name Communications Server für Linux für beide Versionen verwendet. Der Begriff CS-Linux-Computer bezeichnet einen beliebigen Computer, auf dem Communications Server für Linux ausgeführt wird. Dies gilt, soweit nicht explizit Unterschiede beschrieben sind.

Dieses Handbuch gilt für Version 6.2.3 von Communications Server für Linux.

---

## Zielgruppe

Dieses Handbuch ist für Systemadministratoren und Anwendungsprogrammierer bestimmt, die mit Communications Server für Linux arbeiten.

### **Systemadministratoren**

Zum Aufgabenbereich eines Systemadministrators gehören die Installation von Communications Server für Linux, die Konfiguration des Systems für die Verbindung mit dem Netz und die Systempflege. Systemadministratoren müssen mit der Hardware, auf der Communications Server für Linux ausgeführt wird, und mit dem Betriebssystem Linux vertraut sein. Sie müssen Kenntnisse zum Netz haben, mit dem das System verbunden ist, und SNA-Konzepte im Grundsatz verstehen.

### **Anwendungsprogrammierer**

Anwendungsprogrammierer entwerfen und codieren Transaktions- und Anwendungsprogramme, die mit Hilfe der CS-Linux-Programmierschnittstellen Daten über ein SNA-Netz senden und empfangen. Sie sollten fundierte Kenntnisse zu SNA, dem fernen Programm, mit dem das Transaktions- oder Anwendungsprogramm kommuniziert, sowie zur Programmierungs- und Betriebsumgebung des Betriebssystems Linux haben.

Ausführlichere Informationen zum Schreiben von Anwendungsprogrammen finden Sie im Handbuch zur jeweiligen API.

## Benutzung des Handbuchs

In diesem Handbuch ist die Aktivierung, Konfiguration und Verwaltung von Communications Server für Linux erläutert.

### Aufbau des Handbuchs

Dieses Handbuch ist wie folgt aufgebaut:

- Kapitel 1, „SNA-Begriffe und -Konzepte“, auf Seite 1 gibt einen Überblick über SNA- und APPN-Konzepte (Advanced Peer-to-Peer Networking).
- Kapitel 2, „Communications Server für Linux verwalten“, auf Seite 31 beschreibt die CS-Linux-Verwaltungstools und erläutert, wie die Konfiguration von Communications Server für Linux vorbereitet, die Software Communications Server für Linux auf einem Server aktiviert und inaktiviert und das Motif-Verwaltungsprogramm bzw. das Befehlszeilenverwaltungsprogramm verwendet wird.
- In Kapitel 3, „Grundlegende Konfigurationsaufgaben“, auf Seite 57 ist die Ausführung grundlegender Konfigurationsaufgaben für CS-Linux-Server erläutert. Dazu gehört das Konfigurieren des Client-Server-Betriebs, das Konfigurieren des SNA-Knotens und der Nachrichtenprotokollierung für Communications Server für Linux.
- In Kapitel 4, „Konnektivitätskomponenten definieren“, auf Seite 63 ist das Konfigurieren der Konnektivität für den CS-Linux-Knoten erläutert.
- Kapitel 5, „Abhängige LUs konfigurieren“, auf Seite 81 enthält eine Beschreibung der Konfiguration abhängiger LUs (Logical Units, logischer Einheiten) für die LU-Typen 0–3 und für LU-Pools.
- Kapitel 6, „APPC-Kommunikation konfigurieren“, auf Seite 85 enthält eine Beschreibung der Konfiguration von APPC (Advanced Program-to-Program Communications).
- In Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 ist die Konfiguration von Benutzeranwendungen erläutert.
- In Kapitel 8, „Durchgriffsdienste konfigurieren“, auf Seite 113 ist erläutert, wie Durchgriffsdienste konfiguriert werden, die die Kommunikation zwischen nicht direkt verbundenen Hostsystemen und lokalen Systemen unterstützen.
- Kapitel 9, „Communications Server für Linux mit NetView verwalten“, auf Seite 127 beschreibt die Verwendung der Funktion für ferne Befehle (RCF, Remote Command Facility) von Communications Server für Linux für die Verwaltung von Communications Server für Linux und das Ausführen von Befehlen auf einem CS-Linux-Knoten über einen NetView-Host.
- In Kapitel 10, „Client-Server-Systeme mit Communications Server für Linux verwalten“, auf Seite 137 ist die Konfiguration und Verwaltung von IBM Remote API Clients erläutert.
- Anhang A, „Planungsblätter für die Konfiguration“, auf Seite 165 enthält Arbeitsblätter zur Konfiguration von Communications Server für Linux.
- Anhang B, „Aufrufbares TP mit Befehlszeilendienstprogramm konfigurieren“, auf Seite 189 enthält Informationen zu dem Befehlszeilendienstprogramm, mit dem der Benutzer oder Entwickler eines TP-Installationsprogramms ein aufrufbares TP definieren kann.

### Typografische Konventionen

Tabelle 1 auf Seite xiii gibt einen Überblick über die in diesem Handbuch verwendeten typografischen Darstellungen.

Tabelle 1. Typografische Konventionen

Element	Typografiebeispiel
Hervorgehobene Wörter Handbuchtitel	<b>Sichern Sie Dateien vor dem Löschen</b> <i>Communications Server für Linux Verwaltungs-</i> <i>handbuch</i>
Datei- oder Pfadname	<b>/usr/spool/uucp/myfile.bkp</b>
Programm oder Anwendung	<b>snaadmin</b>
Parameter oder Motif-Feld	<i>opcode; LU-Name</i>
Ein Literalwert oder eine Auswahl, den/die der Benutzer eingeben kann (einschließlich Standardwerte)	255; Bei Initialisierung des Knotens
Motif-Schaltfläche	<b>Status</b>
Motif-Menü	<b>Dienste</b>
Motif-Menüeintrag	<b>Knotenparameter konfigurieren</b>
Benutzereingabe	<b>Op1</b>
Computerausgabe	<b>CLOSE</b>
Befehl oder Linux-Dienstprogramm	<b>define_node; cd</b>
Allgemeiner Verweis auf alle Befehle eines bestimmten Typs	<b>query_*</b> (steht für alle Verwaltungsbefehle, die Einzelangaben zu einer Ressource abfragen)
Option	<b>-i</b>
Variable, die einen angegebenen Wert repräsentiert	<i>Dateiname; LU-Name; Benutzer-ID</i>
Rückgabewert	0; -1
3270-Taste	ENTER
Tasten der Tastatur	<b>Strg+D; Eingabetaste</b>
Hexadezimalwert	0x20
Umgebungsvariable	PATH
Funktion, Aufruf oder Eingangspunkt	ioctl
Verb für die Programmierung	GET_LU_STATUS

## Konventionen der grafischen Darstellung

UNIX

Dieses Symbol zeigt den Beginn eines Textabschnitts an, der nur für die Betriebssysteme AIX und Linux gilt. Er bezieht sich auf Linux-Server und den unter AIX, Linux, Linux für pSeries oder Linux für System z ausgeführten IBM Remote API Client.

WINDOWS

Dieses Symbol zeigt den Beginn eines Textabschnitts an, der für den IBM Remote API Client unter Windows gilt.



Dieses Symbol zeigt das Ende eines betriebssystemspezifischen Textes an. Die auf dieses Symbol folgenden Informationen gelten unabhängig vom jeweiligen Betriebssystem.

### Neuerungen in diesem Release

Communications Server für Linux Version 6.2.3 ersetzt Communications Server für Linux Version 6.2.2, Communications Server für Linux Version 6.2.1 und Communications Server für Linux Version 6.2.

Folgende Releases dieses Produkts werden noch unterstützt:

- Communications Server für Linux Version 6.2.2
- Communications Server für Linux Version 6.2.1

Folgende Releases des Produkts werden nicht mehr unterstützt:

- Communications Server für Linux Version 6.2
- Communications Server für Linux Version 6.0.1, verfügbar als PRPQ 5799-RQA oder 5799-RXL.

Communications Server für Linux Version 6.2.3 arbeitet mit IBM Remote API Client Version 6.3.1.0.

### Neue Funktionen

In diesem Release wurde Communications Server für Linux um folgende Funktionen erweitert:

- Neben IPV4-Adressen werden jetzt auch IPV6-Adressen unterstützt.
  - TN Server und Enterprise Extender (HPR/IP) sind auf IP-Konnektivität angewiesen und können unter Verwendung von IPV4 oder IPV6 kommunizieren.
  - Bei einem Client-Server-Deployment können Remote API Clients für die Kommunikation mit Servern IPV4 oder IPV6 verwenden. (IPV6 unterstützt keine UDP-Broadcasts, so dass jeder Client mit mindestens einem Servernamen konfiguriert sein muss.)
  - Die API NOF, das Motif-Verwaltungsprogramm und das Befehlszeilenverwaltungsprogramm akzeptieren IPV6-Hexadezimaladressen mit Doppelpunkt oder IPV4-Adressen in Schreibweise mit Trennzeichen.
- Communications Server für Linux nimmt in eine NOTIFY-Anforderung an den Host für eine TN3270-Sitzung oder für eine LUA-Anwendung auf einem Remote API Client normalerweise den Steuervektor für TCP/IP-Informationen (0x64) auf. Dieser Steuervektor stellt Informationen zum Client bereit, die auf der Hostkonsole angezeigt oder vom Host verwendet werden können (z. B. für die Gebührenabrechnung). Zu diesen Informationen gehört auch die IP-Adresse des Clients. Wenn es sich um eine IPV6-Adresse handelt, der Host jedoch eine frühere Version von VTAM ausführt, die IPV6-Adressen nicht interpretieren kann, wird die Clientadresse möglicherweise nicht richtig auf der Hostkonsole angezeigt. In solchen Fällen können Sie diese Funktion inaktivieren, indem Sie den Parameter *ptf\_flags* des Befehls **define\_node** mit NO\_TCPIP\_VECTOR angeben.
- Das Befehlszeilenverwaltungsprogramm und die API NOF stellen jetzt eine Funktion bereit, mit der die Remote API Clients abgefragt werden können, die derzeit einen bestimmten Server verwenden. Eine NOF-Anwendung kann sich außerdem für den Empfang von Meldungen registrieren, wenn sich Clients eine Verbindung herstellen bzw. trennen.

- Neu hinzugekommen ist die Unterstützung für eine Erreichbarkeitsanzeige des Verbindungsnetzes in Verbindung mit einer ähnlichen Unterstützung auf dem Hostcomputer. Wenn eine einzelne Route einer Transporteinrichtung für gemeinsamen Zugriff (SATE, Shared-Access Transport Facility) nicht verfügbar ist (weil beispielsweise eine Route durch einen IP-Router inaktiviert wurde), werden nach Möglichkeit alternative Routen durch diese Einrichtung verwendet, bevor erneut versucht wird, die ausgefallene Route zu verwenden.
- Durch das Filtern von Protokollen können mehrere Instanzen einer Protokollnachricht unterdrückt werden, so dass jede Nachricht einer angegebenen Liste nur einmal protokolliert wird. Dadurch verringert sich das Informationsvolumen in Protokolldateien, und Sie können sich ganz auf neue oder wichtige Protokollnachrichten konzentrieren.

## Nicht mehr unterstützte Funktionen

In diesem Release wurden keine Funktionen gesperrt.

---

## Referenzliteratur

Informationen zur SNA-, APPN- oder LU-6.2-Architektur finden Sie in den folgenden IBM Dokumenten:

- *IBM System/390 Principles of Operation*, IBM Form SA22-7201
- *IBM z/Architect Principles of Operation*, IBM Form SA22-7832
- *IBM Systemnetzwerkarchitektur*:
  - *LU 6.2 Reference—Peer Protocols*, IBM Form SC31-6808
  - *APPN Architecture Reference*, IBM Form SC30-3422.
  - *Management Services*, IBM Form SC30-3346
  - *Formats*, IBM Form GA27-3136
  - *Technical Overview*, IBM Form GC30-3073

## Referenzliteratur

---

## Kapitel 1. SNA-Begriffe und -Konzepte

Dieses Kapitel enthält Definitionen von Begriffen und Konzepten der Systemnetzwerkarchitektur (SNA, Systems Network Architecture), die eine wichtige Voraussetzung für das Arbeiten mit Communications Server für Linux sind. Communications Server für Linux, die Funktionalität von Communications Server für Linux und die Implementierung der verschiedenen beschriebenen SNA-Konzepte durch Communications Server für Linux sind in der Veröffentlichung *Einstieg in Communications Server für Linux* erläutert. Wenn Sie bereits mit SNA und Communications Server für Linux vertraut sind, können Sie mit Kapitel 2, „Communications Server für Linux verwalten“, auf Seite 31 beginnen.

Das vorliegende Kapitel ist in folgende Abschnitte unterteilt:

- Der Abschnitt „Systemnetzwerkarchitektur“ enthält eine Definition der SNA.
- Im Abschnitt „SNA-Basiskonzepte“ auf Seite 2 werden Begriffe und Konzepte erläutert, die für jedes SNA-Netz gelten.
- Im Abschnitt „APPN-Basiskonzepte“ auf Seite 14 werden Begriffe und Konzepte erläutert, die nur für SNA-Netze mit Unterstützung für das Advanced Peer-to-Peer Networking (APPN) gelten.
- Der Abschnitt „Von APPN-Netzen auf Unterbereichsnetze zugreifen“ auf Seite 30 stellt Begriffe und Konzepte vor, die für kombinierte SNA-APPN-Netze gelten.

**Anmerkung:** In diesem Kapitel können nicht alle zu SNA-Konzepten verfügbaren Informationen bereitgestellt werden. Ausführliche Hinweise zur SNA können Sie den im Abschnitt „Referenzliteratur“ auf Seite xv angegebenen Veröffentlichungen zur SNA entnehmen.

---

### Systemnetzwerkarchitektur

Die Systemnetzwerkarchitektur (SNA, Systems Network Architecture) ist eine Datenübertragungsarchitektur von IBM, die allgemeine Konventionen für die Kommunikation zwischen den verschiedensten Hardware- und Softwareprodukten vorgibt. Diese Architektur umfasst zwei Arten von Definitionen: Formate, die das Layout der zwischen den Netzkomponenten ausgetauschten Nachrichten definieren, und Protokolle, die die Reaktion von Netzkomponenten auf Nachrichten festlegen.

Ein SNA-Netz besteht aus einer Reihe von Computern, die miteinander verbunden sind und über die SNA kommunizieren.

Ursprünglich wurde die SNA für die Kommunikation mit einem Host entwickelt. Alle Netze oder Teilnetze wurden vom Host gesteuert. Die Computer konnten direkt mit dem Host, jedoch nicht untereinander kommunizieren. Dieser ältere hostgesteuerte Netztyp wird häufig als SNA-Unterbereichsnetz bezeichnet. Die SNA wurde seitdem erweitert und unterstützt heute die direkte Peer-zu-Peer-Kommunikation zwischen Computern im Netz, für die kein Host erforderlich ist. Diese neue Art des Netzbetriebs auf Peerebene wird als APPN bezeichnet.

Viele SNA-Netze enthalten Elemente eines Unterbereichsnetzes und eines APPN-Netzes. Bei der Umstellung von einem SNA-Unterbereichsnetz auf ein APPN-Netz kann ein APPN-fähiger Host eingesetzt werden, der sowohl ältere Systeme steuern als auch für neue Systeme als Peer fungieren kann. Einzelne Computer können dann auf Peercomputer (in einem APPN-Netz) und auf einen Host zugreifen. Die

Kommunikation mit dem Host wird vom Host gesteuert, der jedoch keinen Einfluss auf die Peer-zu-Peer-Kommunikation mit anderen Computern hat.

---

### SNA-Basiskonzepte

SNA definiert die Standards, Protokolle und Funktionen, die Einheiten vom Großrechner bis zum Terminal ermöglichen, in SNA-Netzen miteinander zu kommunizieren.

Die SNA-Funktionen sind einer hierarchischen Struktur mit verschiedenen Schichten zugeordnet. Jede dieser Schichten führt eine bestimmte Gruppe von Funktionen aus. Dank der Unterteilung der Netzfunktionen in Schichten können Netzeinheiten Informationen und Verarbeitungsressourcen gemeinsam nutzen, ohne dafür detaillierte Informationen zu den einzelnen Einheiten im Netz zu benötigen. Der Benutzer einer Workstation kann mit einem anderen Benutzer kommunizieren, ohne etwas über die physischen Einheiten im Netz oder die Verbindungen zwischen diesen Einheiten zu wissen.

### Netztypen

SNA unterstützt folgende Netztypen:

- Ein Unterbereichsnetz ist ein hierarchisch organisiertes Netz, das aus Unterbereichsknoten und peripheren Knoten besteht. Unterbereichsknoten wie Hosts und DFV-Controller bearbeiten das allgemeine Routing im Netz. Periphere Knoten wie Terminals stellen eine Verbindung zum Netz her, ohne das allgemeine Routing im Netz zu erkennen.
- Ein Peernetz ist ein kooperativ organisiertes und aus Peerknoten bestehendes Netz, in dem alle Knoten am allgemeinen Routing im Netz teilnehmen.
- Ein gemischtes Netz ist ein Netz, das sowohl die hostgesteuerte Kommunikation als auch die Peerkommunikation unterstützt.

**Anmerkung:** Ein Linux-System, das Communications Server für Linux ausführt, kann in einem Unterbereichsnetz als peripherer Knoten und in einem Peernetz als Peerknoten fungieren oder beide Aufgaben gleichzeitig wahrnehmen.

### SNA-Knoten

In SNA-Netzen ist ein Knoten ein Linux-System oder eine andere Einheit mit zugehörigen Softwarekomponenten, das bzw. die SNA-Protokolle implementiert und über mindestens einen Übertragungsweg zu einem anderen Knoten im Netz verfügt. Jeder Knoten verwaltet sein Ende des Netzübertragungswegs und verwendet SNA-Protokolle, um mit den Knoten am anderen Ende der einzelnen Übertragungswege zu kommunizieren.

Da Unterbereichsnetze und Peernetze die Anordnungsbeziehung zwischen Knoten unterschiedlich definieren, verwenden sie (zur Beschreibung der Rolle von Knoten im Netz) auch verschiedene Bezeichnungen für Knotentypen.

#### Knotentypen in einem Unterbereichsnetz

SNA-Unterbereichsnetze unterstützen folgende Knotentypen:

- Unterbereichsknoten steuern die Kommunikation und die Netzressourcen für alle angeschlossenen Knoten. SNA klassifiziert Unterbereichsknoten nach ihrem Leistungsspektrum und nach dem Umfang, in dem sie andere Knoten steuern:
  - Knoten des Typs 5 stellen SNA-Funktionen für die Steuerung der Netzressourcen sowie Endbenutzerdienste bereit und unterstützen Transaktions-

- programme und Netzbediener. Da diese Funktionen häufig von Hostprozessoren zur Verfügung gestellt werden, werden Knoten des Typs 5 auch als Hostknoten bezeichnet. Die von einem Unterbereichsknoten des Typs 5 gesteuerten Einheiten und Ressourcen bilden die Domäne dieses Knotens.
- Knoten des Typs 4 stellen SNA-Funktionen für das Routing und die Steuerung des Datenflusses in einem Teil des Netzes bereit. Da diese Funktionen häufig von DFV-Controllern zur Verfügung gestellt werden, werden Knoten des Typs 4 auch als Knoten mit DFV-Controller bezeichnet.
  - Periphere Knoten spielen in Unterbereichsnetzen eine untergeordnete Rolle. Ein peripherer Knoten kann beispielsweise eine 3270-Emulation oder die abhängige LU-6.2-Kommunikation unterstützen. Periphere Knoten sind Einheiten wie verteilte Prozessoren, Clustercontroller oder Workstations und werden auch den Knotentypen 2.0 und 2.1 zugeordnet.
    - Knoten des Typs 2.0 werden immer von einem Knoten des Typs 4 oder 5 gesteuert. Ohne einen Knoten des Typs 4 oder 5 können sie keine Übertragung zu anderen Knoten ausführen. Knoten des Typs 2.0 werden als abhängige Knoten bezeichnet.
    - Knoten des Typs 2.1 können als abhängige Knoten eingesetzt werden. Sie können aber auch direkt mit anderen Knoten des Typs 2.1 kommunizieren.

**Anmerkung:** Linux-Computer, auf denen Communications Server für Linux ausgeführt wird, können als Knoten des Typs 2.1 oder 2.0 verwendet werden.

Ein Unterbereichsknoten des Typs 4 oder 5, an den ein peripherer Knoten angeschlossen ist, hat die Rolle eines Grenzknotens. Er führt eine Grenzfunktion aus, indem er die von einem Unterbereichsknoten verwendeten Netzadressen in die von einem peripheren Knoten verwendeten lokalen Adressen umsetzt und umgekehrt.

Ein einfaches Unterbereichsnetz enthält die folgenden Komponenten:

**Host** Ein Host ist ein mit dem Originalsystem IBM System/370 kompatibler Großrechner. Traditionell ist ein Host ein Knoten des Typs 5. Communications Server für Linux auf System z wird jedoch auf einem Host als Knoten des Typs 2.1 oder 2.0 ausgeführt.

### **DFV-Controller**

Ein DFV-Controller wird auch als FEP (Front-End-Prozessor) bezeichnet und ist ein gesondert an den Host angeschlossener Prozessor. Dieser Prozessor verwaltet die Kommunikation des Hosts mit anderen Computern.

### **DFV-Verbindung**

Eine DFV-Verbindung verbindet den Standort des Hosts mit dem Standort eines Endbenutzers. Die Benutzer befinden sich in der Regel an einem anderen Standort als der Host, so dass beide Standorte durch eine DFV-Verbindung miteinander verbunden werden müssen.

### **Terminalcontroller**

Am fernen Ende der DFV-Verbindung befindet sich ein Terminalcontroller, der auch als Clustercontroller bezeichnet wird. Er ist für die Steuerung der Verbindungsverwendung verantwortlich und leitet Daten an die Terminals weiter. Die bekanntesten IBM Terminalcontroller sind IBM 3174 und IBM 3274.

### **Terminals**

Benutzer arbeiten mit Terminals, um Hostanwendungen auszuführen oder

## SNA-Basiskonzepte

Jobs an den Host zu übergeben. Das bekannteste IBM Terminal ist das Datensichtgerät IBM 3270. Ein Terminal kann direkt oder über einen Terminalcontroller mit einem DFV-Controller verbunden sein.

### Drucker

An den Terminalcontroller können auch Drucker wie der Drucker IBM 3287 angeschlossen werden. Die Drucker können Ausgaben vom Host empfangen.

Abb. 1 zeigt, dass die grafische Darstellung eines Unterbereichsnetzes wie ein umgekehrter Baum aussieht.

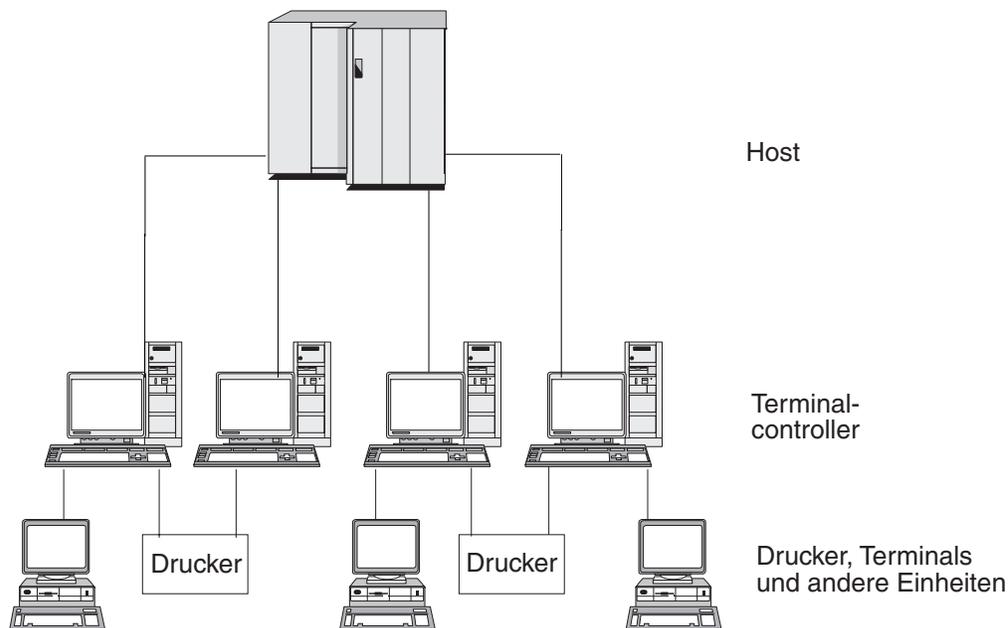


Abbildung 1. SNA-Unterbereichsnetz

Die Wurzel des Baums (oben in der Abbildung) ist der Computer, der das Netz steuert. Die Zweige sind die DFV-Verbindungen vom Host zu den anderen Computern im Netz (Terminalcontroller). Die Blätter (unten in der Abbildung) sind die an diese Computer angeschlossenen Terminals oder Drucker, auf die Benutzer zugreifen.

Der hier beschriebene herkömmliche Aufbau eines SNA-Unterbereichsnetzes ermöglicht den Benutzern, die Ressourcen eines Hostsystems zu verwenden. Die Terminals stellen nur einfache Funktionen für die Eingabe und Anzeige von Daten für den und vom Terminalcontroller bereit. Die SNA-Kommunikation zwischen den Terminals und dem Host wird vom Terminalcontroller gesteuert.

Der Terminalcontroller und die zugehörigen Terminals können durch einen SNA-Knoten ersetzt werden, der ein Produkt wie Communications Server für Linux ausführt. Aus Sicht des Hosts erscheint der Knoten als Terminalcontroller. Er bietet den Benutzern jedoch zusätzliche Funktionen, z. B. für die Anpassung der Bildschirmanzeige, und ermöglicht ihnen den Zugriff auf mehr als ein Hostsystem. Communications Server für Linux wird außerdem auf Linux-Computern ausgeführt, die auch für Aufgaben, die nicht im Zusammenhang mit SNA stehen, verwendet werden können. (Ein Terminalcontroller kann hingegen nur für die Kommunikation mit dem Host eingesetzt werden.)

## Knotentypen in einem Peernetz

Peernetze klassifizieren Knoten nicht hierarchisch, wie es in einem Unterbereichsnetz der Fall ist. Der Datenaustausch mit anderen Knoten wird nicht von einem Host oder einem anderen zentralen Prozessor gesteuert. Ein Knoten kann stattdessen mit jedem beliebigen Knoten im Netz kommunizieren.

Ein Peernetz besteht aus Knoten des Typs 2.1. Die Knoten eines Peernetzes können die folgenden Rollen haben:

- APPN-Netzknoten identifizieren die Position von Netzressourcen, bestimmen Routen für Sitzungen zwischen diesen Ressourcen, leiten Sitzungen weiter und bedienen direkt an den Netzknoten angeschlossene Endknoten und LEN-Knoten. Die Domäne eines APPN-Netzknotens umfasst den Knoten selbst und alle Endknoten, für die dieser Netzverwaltungsdienste bereitstellt.
- APPN-Endknoten können auf ferne Ressourcen zugreifen, ohne dass diese Ressourcen auf dem Endknoten konfiguriert sein müssen. Ein Endknoten kann selbstständig mit Nachbarknoten kommunizieren, benötigt für den Zugriff auf andere Knoten allerdings die Dienste eines Netzknotenservers. Die Domäne eines APPN-Endknotens umfasst nur den Knoten selbst.
- Mit APPN-Zweignetzknoten kann das APPN-Netz in Zweignetze gegliedert werden. Auf diese Weise wird die Topologie des Netzes vereinfacht und der Aufwand für die Netzverwaltung reduziert. APPN-Zweignetzknoten stellen für die Endknoten in einem Zweignetz Netzknotenfunktionen bereit, sind jedoch aus Sicht des Hauptnetzes selbst Endknoten. Weitere Informationen hierzu finden Sie im Abschnitt „Branch Extender“ auf Seite 28.
- LEN-Knoten (Low-Entry Networking) sind Knoten des Typs 2.1, die keine Unterstützung für APPN-Funktionen bieten. Sie können mit Nachbarknoten im APPN-Netz kommunizieren, sind jedoch nicht Bestandteil des APPN-Netzes. Auf einem LEN-Knoten müssen alle potenziellen Sitzungen mit fernen LUs vorab definiert werden. Dies kann in Form von Einzeldefinitionen geschehen oder durch einen Standardeintrag, der angibt, dass sich alle fernen LUs auf einem benachbarten Netzknoten befinden, auf den über eine bestimmte Verbindung zugegriffen werden kann. Die Domäne eines LEN-Knotens umfasst nur den Knoten selbst.

Weitere Informationen zu peerorientierten Knotentypen finden Sie im Abschnitt „Arten von APPN-Knoten“ auf Seite 14.

## Konnektivität

Voraussetzung für die Kommunikation zweier Knoten ist, dass jeder Knoten über eine Kombination von Hardware und Software verfügt, die den Datenfluss zwischen den Knoten unterstützt. Die Hardwarekomponente umfasst einen Adapter pro Knoten und das die beiden Adapter verbindende Übertragungsmedium. Die Softwarekomponente übernimmt die Steuerung der Hardware und der ausgetauschten Daten.

Jeder mit einem Netz verbundene Knoten verfügt über eine oder mehrere Verbindungsstation(en). Verbindungsstationen sind die Hardware- und Softwarekomponenten eines Knotens, die den Datenfluss zu einem bestimmten Nachbarknoten steuern. Zwei Knoten können erst miteinander kommunizieren, wenn eine der Verbindungsstationen die Verbindung zwischen den Knoten aktiviert hat.

## Transaktionsprogramme

Programme, die Informationen über das SNA-Netz austauschen, werden Transaktionsprogramme (TPs, Transaction Programs) genannt.

Nachfolgend sind einige Anwendungen aufgeführt, die SNA-Transaktionsprogramme enthalten können:

- Emulationsprogramme
- Dateiübertragung
- Datenbanktransaktionsverarbeitung
- Netzverwaltung
- zentrale Datenbankdienste

Das Transaktionsprogramm greift über eine logische Einheit (LU, Logical Unit) auf das Netz zu. Die LU baut eine Sitzung mit einer Partner-LU auf einem anderen Knoten auf und verwaltet diese Sitzung. Weitere Informationen zu logischen Einheiten finden Sie im Abschnitt „Logische Einheiten“ auf Seite 7.

**Anmerkung:** Zu Communications Server für Linux werden Beispiel-TPs für die meisten unterstützten APIs geliefert. Weitere Informationen zu Beispiel-TPs können Sie dem Programmierhandbuch zur jeweiligen API entnehmen. Sie können auch zu anderen Produkten gehörende SNA-Transaktionsprogramme kaufen oder eigene Transaktionsprogramme erstellen (siehe Abschnitt „Anwendungsprogrammierschnittstellen“).

## Anwendungsprogrammierschnittstellen

Für das Erstellen von SNA-Transaktionsprogrammen werden Anwendungsprogrammierschnittstellen (APIs, Application Programming Interfaces) verwendet. APIs stellen bestimmte Unterroutinen bereit, über die SNA-Transaktionsprogramme auf SNA-Funktionen, z. B. auf Funktionen für den Datenaustausch und Steuerfunktionen, zugreifen können. Mit Hilfe dieser Unterroutinen kann ein SNA-Transaktionsprogramm mit einem anderen SNA-Transaktionsprogramm auf einem fernen Knoten kommunizieren.

Zu Communications Server für Linux gehören die folgenden APIs auf allen Plattformen:

- APPC, nur LU 6.2
- CPI-C (Common Programming Interface for Communications), nur LU 6.2
- API CSV (Common Service Verb)
- API LUA

Im Lieferumfang von Communications Server für Linux sind außerdem die folgenden privaten Programmierschnittstellen enthalten:

- API MS (Management Services), nur für AIX- oder Linux-Systeme
- API NOF (Node Operator Facility)

## Adressierbare Netzeinheiten

Die Kommunikation zwischen einem Transaktionsprogramm und dem SNA-Netz erfolgt über adressierbare Netzeinheiten (NAUs, Network Accessible Units). Diese Einheiten sind eindeutige Netzressourcen, auf die andere Netzressourcen (unter Angabe eindeutiger lokaler Adressen) zugreifen können.

SNA stellt die folgenden Arten von NAUs bereit:

- physische Einheiten (siehe Abschnitt „Physische Einheiten“ auf Seite 7)
- logische Einheiten (siehe Abschnitt „Logische Einheiten“ auf Seite 7)

- Steuerpunkte (siehe Abschnitt „Steuerpunkte“ auf Seite 8)

**Anmerkung:** Da Transaktionsprogramme als Benutzer des Netzes und nicht als Netzkomponenten angesehen werden, sind sie nicht den NAUs zuzuordnen.

### Physische Einheiten

Jeder SNA-Knoten enthält eine physische Einheit (PU, Physical Unit). Die PU verwaltet Ressourcen (z. B. Verbindungsressourcen) und unterstützt die Kommunikation mit einem Host.

**Anmerkung:** Auf Knoten des Typs 2.1 (z. B. APPN-Knoten) stellt der Steuerpunkt neben anderen Diensten auch PU-Dienste bereit (siehe Abschnitt „Steuerpunkte“ auf Seite 8). Zwei Knoten des Typs 2.1 (z. B. CS-Linux-Knoten) können direkt und ohne die Dienste eines Hosts miteinander kommunizieren.

### Logische Einheiten

Jeder SNA-Knoten enthält eine oder mehrere logische Einheit(en) (LUs, Logical Units). Eine LU stellt eine Reihe von Funktionen bereit, die Transaktionsprogrammen und Endbenutzern den Zugriff auf das Netz ermöglichen. LUs kommunizieren direkt mit lokalen Transaktionsprogrammen und Einheiten.

SNA definiert LUs verschiedener Typen. Jeder LU-Typ ist für eine bestimmte Klasse von Anwendungen optimiert. LUs unterschiedlicher Typen können nicht miteinander kommunizieren. LUs desselben Typs können jedoch selbst dann miteinander kommunizieren, wenn sie sich auf Systemen verschiedener Typen befinden.

Ein Transaktionsprogramm auf einem Linux-System kann beispielsweise ebenso einfach mit einem Transaktionsprogramm auf einer AS/400 wie mit einem Transaktionsprogramm auf einem anderen Linux-System kommunizieren, wenn nur beide Transaktionsprogramme LUs desselben Typs verwenden.

Communications Server für Linux unterstützt folgende LU-Typen:

#### LU 6.2 (für APPC, 5250, APPC Application Suite und CPI-C)

LUs des Typs 6.2 unterstützen die Kommunikation zwischen Programmen in einer verteilten Datenverarbeitungsumgebung. Der LU-6.2-Datenstrom ist entweder ein allgemeiner SNA-Datenstrom (ein feldstrukturierter Datenstrom) oder ein benutzerdefinierter Datenstrom. LUs 6.2 können für die Kommunikation zwischen zwei Knoten des Typs 5, einem Knoten des Typs 5 und einem Knoten des Typs 2.0 oder 2.1 oder zwei Knoten des Typs 2.1 verwendet werden. (Knoten des Typs 2.1 können als APPN-Knoten eingesetzt werden.)

LUs dieses Typs bieten mehr Funktionen und eine höhere Flexibilität als LUs aller anderen Typen. Sofern Sie nicht durch vorhandene Hardware oder Software eingeschränkt sind, sollten Sie für die Entwicklung neuer Anwendungen LUs des Typs 6.2 verwenden.

**Anmerkung:** Nur LUs 6.2 können unabhängige LU-Funktionen bereitstellen.

#### LU 3 (für 3270-Druck)

LUs des Typs 3 unterstützen Anwendungsprogramme und Drucker mit Hilfe des SNA-3270-Datenstroms.

Eine LU 3 kann beispielsweise ein unter CICS (Customer Information Control System) ausgeführtes Programm unterstützen, das Daten an einen an den Establishment Controller IBM 3174 angeschlossenen Drucker IBM 3262 sendet.

### LU 2 (für 3270-Terminals)

LUs des Typs 2 unterstützen Anwendungsprogramme und Terminals mit Hilfe des SNA-3270-Datenstroms. LUs des Typs 2 nutzen den SNA-3270-Datenstrom auch für die Dateiübertragung.

Das LU-2-Protokoll kann beispielsweise 3270-Emulationsprogramme unterstützen, die es Workstations erlauben, die Funktionen von Terminals der Produktfamilie IBM 3270 auszuführen. Andere Programme verwenden LUs des Typs 2 für die Kommunikation mit Hostanwendungen, die normalerweise Ausgaben für 3270-Terminals liefern. Über solche Transaktionsprogramme kann die Workstation eine Art Verbundverarbeitung mit dem Host erreichen.

### LU 1 (für SCS-Druck und RJE)

LUs des Typs 1 unterstützen Anwendungsprogramme und eine oder mehrere Einheit(en) umfassende Datenverarbeitungsstationen, die in einer interaktiven oder verteilten Datenverarbeitungs Umgebung bzw. in einer Umgebung mit Datenübertragung im Stapelbetrieb kommunizieren. Die von LUs des Typs 1 verwendeten Datenströme gehen mit der SNA-Zeichenfolge oder der DCA (Document Content Architecture) konform.

LUs des Typs 1 können beispielsweise ein Anwendungsprogramm unterstützen, das unter IMS/VS (Information Management System/Virtual Storage) ausgeführt wird und mit einem Informationssystem IBM 8100 kommuniziert. Dadurch ermöglichen sie dem Bediener, eine vom Anwendungsprogramm verwaltete Datenbank zu korrigieren.

Anwendungen, die LUs des Typs 1 verwenden, werden oft als Anwendungen für Jobferneingabe (RJE-Anwendungen, Remote Job Entry) bezeichnet.

### LU 0 (für LUA)

Die ältere LU-Definition LU 0 unterstützt eine primitive Kommunikation zwischen Programmen. LU 0 wird von bestimmten Hostdatenbanksystemen wie IMS/VS (Information Management System/Virtual Storage) und einigen Systemen in Filialen des Einzelhandels und des Bankwesens (z. B. des Handelssystems IBM 4680) verwendet. Aktuellere Releases dieser Produkte unterstützen ebenfalls die LU-6.2-Kommunikation, der für neue Anwendungen der Vorzug zu geben ist.

**Anmerkung:** Informationen zu den von logischen SNA-Einheiten verwendeten Datenströmen finden Sie in der Veröffentlichung *Systems Network Architecture Technical Reference*.

## Steuerpunkte

Ein Steuerpunkt (CP, Control Point) ist eine NAU, die Netzressourcen innerhalb ihrer Domäne verwaltet, indem sie die Aktivierung und Inaktivierung von Ressourcen und die Statusüberwachung steuert. Der CP verwaltet physische Ressourcen wie Verbindungen und logische Informationen wie Netzadressen.

SNA definiert die folgenden Arten von Netzsteuerpunkten:

### Steuerpunkt für Systemdienste

Der CP eines Knotens vom Typ 5 wird als Steuerpunkt für Systemdienste (SSCP, System Services Control Point) bezeichnet. Er verwaltet und steuert die Netzressourcen in einem Unterbereichsnetz. Ein SSCP kann beispiels-

weise eine von ihm gesteuerte LU in einem Verzeichnis mit Netzressourcen lokalisieren und die Kommunikation zwischen zwei LUs in seiner Domäne einrichten. Ein SSCP kann auch mit anderen SSCPs zusammenarbeiten, um Konnektivität zwischen LUs in verschiedenen Unterbereichsdomänen herzustellen.

Der SSCP stellt für Netzbediener am Hostsystem eine Schnittstelle zum Prüfen und Steuern der Ressourcen im Netz bereit.

### **Steuerpunkt der physischen Einheit**

Der Steuerpunkt von Knoten des Typs 4 und des Typs 2.0 in einem Unterbereichsnetz wird als Steuerpunkt der physischen Einheit (PUCP, Physical Unit Control Point) bezeichnet.

### **Steuerpunkt**

Auf Knoten des Typs 2.1 stellt der Steuerpunkt sowohl PU- als auch LU-Funktionen, z. B. für die Aktivierung von Verbindungsstationen, für die Interaktion mit einem lokalen Bediener und die Verwaltung lokaler Ressourcen, bereit. Er bietet aber auch Netzdienste wie die Lokalisierung von Partner-LUs und die Routenauswahl für alle lokalen LUs an.

In einem Unterbereichsnetz kann der CP eines CS-Linux-Knotens als PU des Typs 2.0 agieren. Er kommuniziert mit dem SSCP auf einem Host, jedoch nicht mit anderen CPs im Unterbereichsnetz.

Ist der CP Teil eines APPN-Netzes, tauscht er mit den CPs der Nachbarknoten Informationen zur Netzsteuerung aus. Der CP kann auch die Funktion einer unabhängigen LU des Typs 6.2 übernehmen. Für Transaktionsprogramme auf dem lokalen Knoten ist der CP die Standard-LU. Weitere Informationen zum APPN-Steuerpunkt finden Sie im Abschnitt „APPN-Steuerpunkt“ auf Seite 16.

## **Sitzungen**

NAUs kommunizieren mit NAUs anderer Knoten über temporäre logische DfV-Kanäle, die als Sitzungen bezeichnet werden. Zwei Transaktionsprogramme können erst miteinander kommunizieren, wenn ihre LUs eine Sitzung aufgebaut haben. Die LU, die die Sitzung auf dem lokalen Knoten verwaltet, ist die lokale LU. Die LU, die die Sitzung auf dem fernen Knoten verwaltet, ist die Partner-LU.

### **Sitzungstypen**

Communications Server für Linux ist im wesentlichen für die folgenden Sitzungstypen konzipiert:

#### **LU-LU-Sitzungen**

Bevor zwei Transaktionsprogramme miteinander kommunizieren können, müssen die diese Transaktionsprogramme unterstützenden LUs eine LU-LU-Sitzung aufbauen. Im Allgemeinen wird eine Sitzung aufgebaut, wenn ein Transaktionsprogramm auf einem SNA-Knoten versucht, mit einem Transaktionsprogramm auf einem anderen Knoten zu kommunizieren, und zwischen den LUs der beiden Knoten keine Sitzung verfügbar ist.

#### **SSCP-LU-Sitzungen**

Eine abhängige LU (siehe Abschnitt „Abhängige und unabhängige LUs“ auf Seite 10) muss zunächst eine aktive SSCP-LU-Sitzung mit einem SSCP auf einem Knoten des Typs 5 haben, bevor sie eine Sitzung mit einer LU im Unterbereichsnetz aufbauen kann. Sobald eine SSCP-LU-Sitzung aktiv ist, kann die abhängige LU eine LU-LU-Sitzung anfordern.

### SSCP-PU-Sitzungen

Eine SSCP-LU-Sitzung kann erst aufgebaut werden, wenn die die LU steuernde PU mit einem SSCP auf einem Knoten des Typs 5 eine aktive SCP-PU-Sitzung ausführt. Über die SSCP-PU-Sitzung werden Steuerdaten und Netzverwaltungsdaten zwischen der PU und dem SSCP ausgetauscht.

### CP-CP-Sitzungen

In einem APPN-Netz können Nachbarknoten CP-CP-Sitzungen aufbauen. Diese Sitzungen werden verwendet, um im APPN-Netz nach Ressourcen zu suchen und Topologiedaten zu verwalten (siehe Abschnitt „APPN-Steuerpunkt“ auf Seite 16).

## Attribute logischer Einheiten für Sitzungen

Logische Einheiten haben Attribute, die ihre Interaktion in LU-LU-Sitzungen festlegen. Diese Attribute werden von der Systemnetzwerkarchitektur bestimmt. LUs können primär oder sekundär und abhängig oder unabhängig sein.

**Primäre und sekundäre LUs:** Zum Aufbauen einer Sitzung fordert eine LU bei der anderen LU die Aktivierung einer Sitzung an, indem sie eine BIND-Anforderung an diese LU sendet.

- Eine primäre LU ist die LU, die für eine gegebene LU-LU-Sitzung die BIND-Anforderung sendet.
- Eine sekundäre LU ist die LU, die die BIND-Anforderung empfängt.

In Peernetzen gibt es keine feste Knotenhierarchie und keine vorab festgelegten primären oder sekundären LUs.

**Anmerkung:** Eine unabhängige LU nimmt in einem Peernetz an mehreren Sitzungen teil (siehe Abschnitt „Mehrfachsitzungen und Parallelsitzungen“ auf Seite 11) und kann für eine Sitzung die Rolle einer primären und für eine andere Sitzung die Rolle einer sekundären LU übernehmen.

**Abhängige und unabhängige LUs:** Alle LUs der Typen 0, 1, 2 und 3 sind abhängige LUs. LUs des Typs 6.2 können als abhängige oder unabhängige LUs konfiguriert werden.

- Eine abhängige LU (die auch als SSCP-abhängige LU bezeichnet wird) benötigt zum Aufbauen einer Sitzung mit einer anderen LU die Dienste eines SSCP. Voraussetzung für das Aufbauen einer LU-LU-Sitzung ist, dass eine SSCP-LU-Sitzung aufgebaut wurde.

Eine abhängige LU kann nur Sitzungen mit LUs auf einem SNA-Host ausführen. Aufgrund dieser Einschränkung verwenden LUs in der Regel Unterbereichsnetze (die auch als hostvermittelte Netze bezeichnet werden). Mit der Funktion DLUR (Dependent LU Requester) kann der Sitzungsverkehr von abhängigen LUs jedoch über APPN-Netze transportiert werden. Weitere Informationen zu DLUR finden Sie im Abschnitt „Von APPN-Netzen auf Unterbereichsnetze zugreifen“ auf Seite 30.

Eine abhängige LU auf einem peripheren Knoten ist immer eine sekundäre LU.

- Eine unabhängige LU kann mit Hilfe eines SNA-Hosts Sitzungen mit anderen unabhängigen LUs aufbauen. LUs des Typs 6.2 sind die einzigen LUs, die unabhängig sein können.

Eine unabhängige LU kann beim Aufbauen einer Sitzung die Rolle einer primären oder einer sekundären LU übernehmen.

## Mehrfachsitzungen und Parallelsitzungen

Eine unabhängige LU kann gleichzeitig an Sitzungen mit mehreren fernen LUs teilnehmen (Mehrfachsitzung).

Eine unabhängige LU kann aber auch an Parallelsitzungen oder mehreren gleichzeitig ablaufenden Sitzungen mit einer fernen LU teilnehmen.

Abhängige LUs (auch abhängige LUs des Typs 6.2) können keine Mehrfachsitzungen ausführen.

Abb. 2 zeigt LUs mit Mehrfach- und Parallelsitzungen. LUA und LUB führen Parallelsitzungen aus. LUA nimmt außerdem an Mehrfachsitzungen teil: zwei mit LUB und eine mit LUD. LUD hat Mehrfachsitzungen mit LUA und LUC aufgebaut.

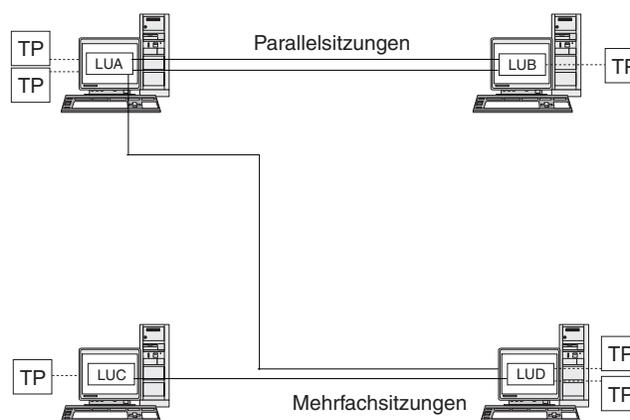


Abbildung 2. Mehrfachsitzungen und Parallelsitzungen

## Dialoge

Dieser Abschnitt gilt nur für LU 6.2.

Wenn zwischen zwei LUs eine Sitzung aufgebaut ist, unterstützt die LU-LU-Sitzung den Datenaustausch zwischen zwei Transaktionsprogrammen, die die Sitzung exklusiv zum Ausführen einer Transaktion verwenden. Dieser Datenaustausch wird als Dialog bezeichnet. Über eine bestimmte Sitzung ist immer nur ein Dialog zur Zeit möglich. Eine Sitzung kann zwar für mehrere Dialoge genutzt werden, diese müssen jedoch nacheinander stattfinden.

Zum Einleiten eines Dialogs sendet ein Quelltransaktionsprogramm eine Anforderung an seine LU und weist die LU an, einen Dialog mit einem fernen Transaktionsprogramm einzurichten. Das aufrufende Transaktionsprogramm (oder Quellen-TP) leitet den Dialog ein, wie ein anrufender Teilnehmer ein Telefongespräch einleitet. Das aufrufbare Transaktionsprogramm oder Ziel-TP (das ferne Transaktionsprogramm) ist der Dialogpartner, der mit dem Teilnehmer, der einen Telefonanruf annimmt, vergleichbar ist.

Abb. 3 auf Seite 12 zeigt den Datenaustausch zwischen Transaktionsprogrammen und LUs, der die Kommunikation zwischen zwei Knoten ermöglicht. Obwohl es den Anschein hat, als würden die Transaktionsprogramme direkt kommunizieren, sind die LUs der Knoten bei jedem Datenaustausch die Vermittler.

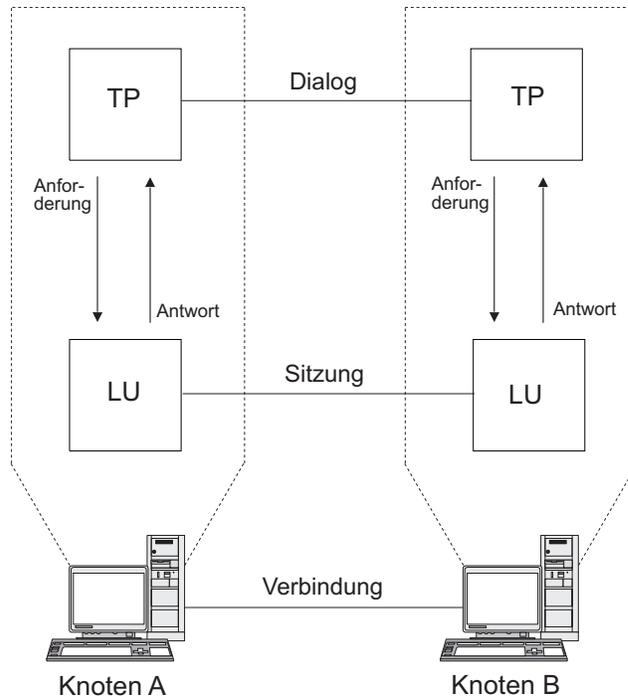


Abbildung 3. Kommunikation zwischen Transaktionsprogrammen und logischen Einheiten

SNA definiert zwei Arten des Datenaustauschs, den formatfreien Austausch und den Austausch formatierter Daten. Diese beiden Arten des Datenaustauschs zeigen mit unterschiedlichen Methoden die Länge der übertragenen oder empfangenen Datenpakete an, die zwischen Communications Server für Linux und dem Transaktionsprogramm übergeben werden.

- Beim Austausch formatierter Daten müssen die Daten vom Transaktionsprogramm vor der Übergabe an die Funktion SEND als logische Sätze formatiert werden.

Ein logischer Satz besteht aus einem Header mit zwei oder vier Bytes, der mit einem 2-Byte-Längenfeld beginnt und oft als „LL“ dargestellt wird, und bis zu 32.765 Bytes Daten. Logische Sätze können zusammengefasst und als Block gesendet werden. Auf diese Weise ist es möglich, mit nur einem Aufruf mehr als einen logischen Satz an die Funktion SEND zu übertragen.

- Beim formatfreien Datenaustausch werden die Informationen als Zeiger auf einen nicht formatierten Datenblock an die Funktion SEND übergeben. Die Länge des Blocks wird durch Übergabe eines gesonderten Parameters mitgeteilt. Der Block kann nicht in Form logischer Sätze empfangen werden. Die erforderliche Formatierung auf Satzebene muss das empfangende Transaktionsprogramm übernehmen.

## Modi

Jeder LU-LU-Sitzung ist ein Modus zugeordnet, der eine Reihe von Sitzungskenndaten definiert. Zu diesen Sitzungskenndaten gehören Parameter für die Nachrichtendosierung, Sitzungsbegrenzungen (wie die maximale Anzahl von Sitzungen zwischen zwei LUs), Größe der Nachrichten und Routing-Parameter.

Jeder Modus ist durch einen eindeutigen Modusnamen gekennzeichnet. Der Name eines Modus muss auf allen SNA-Knoten, die diesen Modus verwenden, identisch sein.

## Routenauswahl

Zum Aufbau einer LU-LU-Sitzung muss zwischen den Knoten, auf denen sich die LUs befinden, eine Route berechnet werden. Eine Route ist eine geordnete Folge von Verbindungen und Knoten, die zwischen den beiden Knoten einen Pfad bilden.

SNA-Netze unterstützen die folgenden Methoden der Routenauswahl:

- In Unterbereichsnetzen müssen Sie alle Routen zwischen Unterbereichsknoten vorab definieren.
- In Peernetzen ohne APPN-Unterstützung können Knoten des Typs 2.1 nur Sitzungen mit Nachbarknoten unterstützen. Sitzungen dieser Knoten können nicht über Zwischenknoten weitergeleitet werden.
- In APPN-Netzen kann SNA zum Zeitpunkt der Sitzungsinitialisierung dynamisch Routen berechnen. Hierfür wird die Serviceklasse verwendet, die für den von der Sitzung benutzten Modus angegeben ist (siehe Abschnitt „Serviceklasse“).

Das APPN-Feature HPR (High-Performance Routing) stellt folgende Funktionen bereit:

- Das Rapid Transport Protocol (RTP) minimiert den Zyklus- und Speicherbedarf für das Routing von Paketen der Vermittlungsschicht über Zwischenknoten auf einer Sitzungsrouten.
- Mit Hilfe von ANR (Automatic Network Routing, automatisches Routing im Netz) können APPN-Netze Sitzungen automatisch weiterleiten, falls ein Bereich der ursprünglich berechneten Route ausfällt.

## Serviceklasse

Die Serviceklasse (COS, Class Of Service) ist eine Definition der Kenndaten eines Transportnetzes (mit Datenübertragungssteuerung und Pfadsteuerung) wie Routensicherheit, Übertragungspriorität und Bandbreite, die der lokale Knoten zum Aufbau einer bestimmten Sitzung verwenden kann. Die COS-Definition ordnet Faktoren wie der Sicherheitsstufe, den Kosten pro Byte, den Kosten pro Verbindungsdauer, der Laufzeitverzögerung und der effektiven Kapazität relative Werte zu.

In einem Unterbereichsnetz wird eine Serviceklasse aus dem Modus abgeleitet, der einer Sitzung laut Definition auf dem Hostsystem zugeordnet ist.

APPN-Netzknoten berechnen mit Hilfe der Serviceklasse Sitzungsrouten zwischen unabhängigen LUs. Weitere Informationen zum Sitzungs-Routing in APPN-Netzen finden Sie im Abschnitt „Sitzungs-Routing“ auf Seite 21.

### APPN-Basiskonzepte

Advanced Peer-to-Peer Networking (APPN) ist eine Netzarchitektur, die die verteilte Netzsteuerung unterstützt. Sie vereinfacht das Konfigurieren von Netzen und das Arbeiten mit Netzen, stellt eine zentrale Netzverwaltung bereit und unterstützt eine flexible Konnektivität.

Ein APPN-Netz besteht aus Knoten des Typs 2.1. Jeder Knoten im Netz ist mit mindestens einem anderen Knoten im APPN-Netz verbunden. Über jede dieser Verbindungen zu Nachbarknoten (Knoten im selben Netz, die direkte Verbindungen ohne Verwendung eines dritten Knotens herstellen können) werden CP-CP-Sitzungen aufgebaut. Alle Knoten in einem APPN-Netz haben einen gemeinsamen Netznamen.

APPN-Knoten können Prozessoren unterschiedlicher Größe sein wie das Application System/400 (AS/400), PCs mit CS/NT, Systeme mit Virtual Terminal Access Method (VTAM) und Linux-Server, auf denen Communications Server für Linux ausgeführt wird.

APPN stellt folgende Funktionen bereit:

- APPN-Steuerpunktfunktionen (siehe Abschnitt „APPN-Steuerpunkt“ auf Seite 16)
- Verzeichnisdienste zur Unterstützung der Suche nach bestimmten logischen Einheiten (siehe Abschnitt „Ressourcen lokalisieren“ auf Seite 17)
- Topologie- und Routing-Dienste, die mit Hilfe der Weitervermittlung einer Sitzungsrouten (ISR, Intermediate Session Routing), dem Automatic Network Routing (ANR) und von Verbindungsnetzen das Aufbauen von Sitzungen unterstützen (siehe Abschnitt „Sitzungs-Routing“ auf Seite 21 und „APPN-Verbindungsnetze“ auf Seite 27)

**Anmerkung:** Ein APPN-Knoten kann auch mit einem Unterbereichsnetz verbunden sein und so in einem Peernetz die Rolle eines APPN-Knotens und in einem Unterbereichsnetz die Rolle eines peripheren Knotens übernehmen.

### Arten von APPN-Knoten

Ein APPN-Netz kann die folgenden Arten von Knoten enthalten:

- Netzknoten (siehe Abschnitt „APPN-Netzknoten“ auf Seite 15)
- Endknoten (siehe Abschnitt „APPN-Endknoten“ auf Seite 16)

Mit einem APPN-Netz können auch LEN-Knoten (Low-Entry Networking) verbunden sein, die jedoch keine APPN-Features nutzen (siehe Abschnitt „LEN-Knoten“ auf Seite 16).

Abb. 4 auf Seite 15 zeigt ein APPN-Beispielnetz.

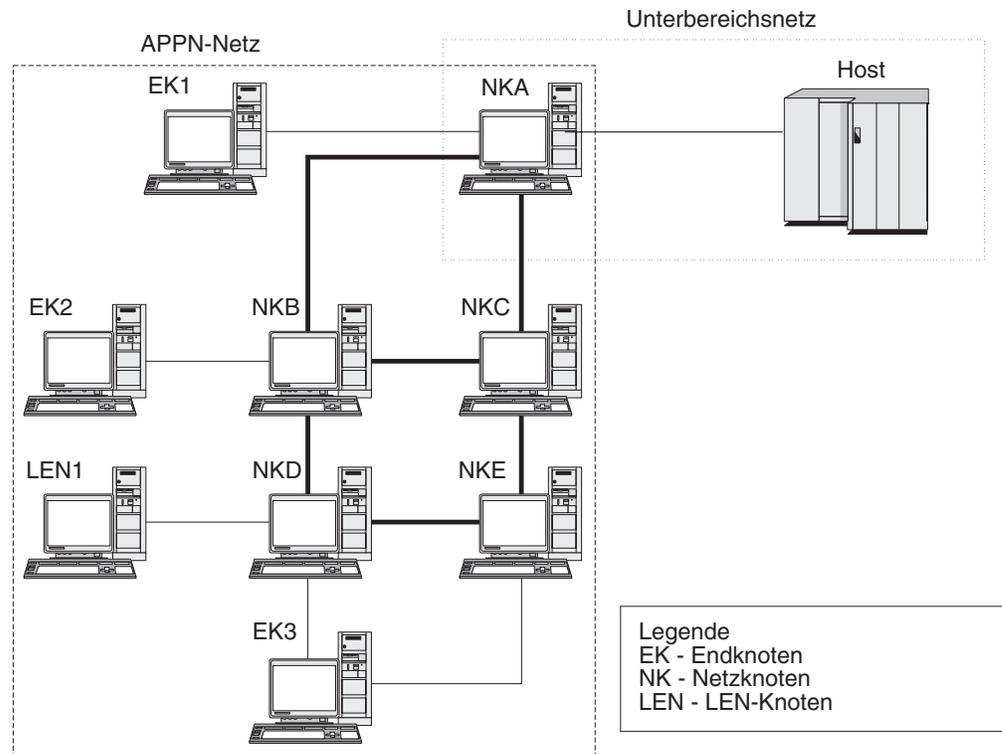


Abbildung 4. Ausschnitt aus einem APPN-Beispielnetz

Dieses Beispiel zeigt ein APPN-Netz mit fünf Netzknotten, drei Endknoten und einem LEN-Knoten. Die Netzknotten bilden die Zentralverbindung des APPN-Netzes. Die Endknoten haben über die Netzknotten Zugang zum Netz. LU-6.2-Transaktionsprogramme jedes Knotens können mit jedem beliebigen anderen LU-6.2-Transaktionsprogramm im Netz kommunizieren.

Einer der APPN-Netzknotten (NKA) ist über einen DFV-Controller mit einem Host verbunden und damit gleichzeitig Teil eines Unterbereichsnetzes. Dieser Knoten ist in der Kommunikation mit den Knoten im APPN-Netz ein APPN-Knoten und in der Kommunikation mit den Knoten im Unterbereichsnetz ein peripherer Knoten. Über diesen Netzknotten können LUs des Typs 6.2 auf anderen Knoten im APPN-Netz LU-LU-Sitzungen mit LUs des Typs 6.2 auf dem Host aufbauen.

### APPN-Netzknotten

Ein APPN-Netzknotten ist ein Knoten des Typs 2.1, der für alle LUs seiner Domäne verteilte Verzeichnis- und Routing-Dienste bereitstellt. Diese LUs können sich auf dem Netzknotten selbst oder auf einem APPN-Endknoten bzw. einem LEN-Knoten befinden, für den der Netzknotten Dienste zur Verfügung stellt. Da ein APPN-Netzknotten für Endknoten und LEN-Knoten seiner Domäne der Netzeingangspunkt ist, wird er auch als Netzknottenserver für diese Knoten bezeichnet.

Ein Netzknotten stellt folgende Dienste bereit:

- LU-LU-Sitzungsdienste für seine lokalen LUs
- Verzeichnisdurchsuchung und Routenauswahl für alle LUs seiner Domäne
- Weitervermittlung einer Sitzungsrute (siehe Abschnitt „Zwischenpfadsteuerung“ auf Seite 24)
- Routing von Verwaltungsdaten wie Alerts zwischen einem bedienten Endknoten und einem zentralen Alert-Verarbeitungssystem.

### APPN-Endknoten

Ein APPN-Endknoten ist ein Knoten des Typs 2.1, der in einem APPN-Netz die Funktion eines Endknotens hat. Der Knoten verwaltet Verzeichnisinformationen nur für lokale Ressourcen. Ein APPN-Endknoten kann selbstständig Sitzungen zwischen lokalen LUs und LUs auf Nachbarknoten aufbauen. Für Sitzungen mit LUs auf nicht direkt mit dem Endknoten verbundenen Knoten fordert der Endknoten über CP-CP-Sitzungen Routing- und Verzeichnisinformationen von seinem Netzknottenserver an.

APPN-Endknoten können ihre lokalen LUs auf ihrem Netzknottenserver registrieren lassen. Dank dieser Möglichkeit muss der Netzbediener am Netzknottenserver nicht vorab die Namen aller LUs auf den angeschlossenen Endknoten, für die der Netzknottenserver Dienste bereitstellt, definieren.

Ein APPN-Endknoten kann an mehrere Netzknotten angeschlossen sein (siehe EK3 in Abb. 4 auf Seite 15), jedoch immer nur mit jeweils einem Netzknotten, seinem Netzknottenserver, aktive CP-CP-Sitzungen haben. Die anderen Netzknotten können für den Endknoten nur eine Route bereitstellen oder als Ausweichnetzknottenserver dienen, falls der Hauptnetzknottenserver nicht verfügbar sein sollte.

Ein APPN-Endknoten kann eine direkte Verbindung zu einem anderen APPN-Endknoten oder einem LEN-Knoten haben. CP-CP-Sitzungen zwischen zwei Endknoten sind jedoch nicht möglich.

### LEN-Knoten

Ein LEN-Knoten (Low-Entry Networking) ist ein Knoten des Typs 2.1, der unabhängige LU-6.2-Protokolle verwendet, aber keine CP-CP-Sitzungen unterstützt. Der Knoten kann mit einem APPN-Netzknotten oder -Endknoten verbunden sein, bietet jedoch keine Unterstützung für APPN-Funktionen.

Ein APPN-Netzknotten kann für einen angeschlossenen LEN-Knoten Routing-Dienste bereitstellen. Auf diese Weise kann der LEN-Knoten Teil eines APPN-Netzes sein, ohne dass zwischen dem LEN-Knoten und allen Knoten im APPN-Netz Verbindungsstationen definiert werden müssen.

Wenn der LEN-Knoten Sitzungen mit LUs des APPN-Netzes aufbauen soll, müssen diese LUs auf dem LEN-Knoten so definiert sein, als würden sie sich auf dem Netzknottenserver des LEN-Knotens befinden. Der LEN-Knoten führt Sitzungen mit den LUs auf seinem Netzknottenserver aus. Der Netzknotten leitet die Sitzung über das APPN-Netz zu dem Knoten im Netz weiter, auf dem sich die LU tatsächlich befindet. LUs auf dem LEN-Knoten müssen vorab auf dem Netzknotten, der den LEN-Knoten bedient, definiert werden. LU-Ressourcen von LEN-Knoten können im Gegensatz zu denen von Endknoten nicht auf dem Netzknottenserver registriert werden.

Ein APPN-Endknoten kann keine Route bereitstellen. Wenn die einzige Verbindung eines LEN-Knotens eine Verbindung zu einem APPN-Endknoten ist, kann der LEN-Knoten nur über die direkte Verbindung zwischen den beiden Knoten mit LUs auf dem Endknoten kommunizieren.

### APPN-Steuerpunkt

Ein APPN-Steuerpunkt ist eine Gruppe von Funktionen zur Verwaltung der Knotenressourcen und zur Unterstützung der Funktionen physischer und logischer Einheiten auf einem Knoten des Typs 2.1. Ein APPN-CP steuert lokale Knotenfunktionen (wie das Aktivieren und Inaktivieren von Adaptern oder Verbindun-

gen), stellt Verzeichnis- und Topologieinformationen bereit und hilft LUs bei der Initialisierung und Beendigung von Sitzungen.

Nachbarknoten in einem APPN-Netz nutzen Paare paralleler CP-CP-Sitzungen, um Netzinformationen auszutauschen und Dienste für Verzeichnisinformationen und Routenauswahl zur Verfügung zu stellen. Beide Sitzungen jedes Sitzungspaares müssen aktiv sein, damit die Partner-CPs ihre Interaktion beginnen und fortführen können. Diese Sitzungen werden von verschiedenen Knoten unterschiedlich genutzt:

- Zwischen einem APPN-Netzknoten und jedem benachbarten Netzknoten werden zwei parallele CP-CP-Sitzungen aufgebaut. Über diese CP-CP-Sitzungen werden Verzeichnis-, Topologie- und Verwaltungsinformationen ausgetauscht.
- Zwischen einem APPN-Endknoten und dem Knoten, der dem Endknoten als Server dient, werden zwei parallele CP-CP-Sitzungen aufgebaut. Über diese CP-CP-Sitzungen werden Verzeichnis-, Topologie- und Verwaltungsinformationen ausgetauscht.
- LEN-Knoten unterstützen keine CP-CP-Sitzungen.

Die in CP-CP-Sitzungen bereitgestellten Funktionen hängen wie folgt von der Art der beteiligten Knoten ab:

- Alle CP-CP-Sitzungen führen Suchen in Verzeichnissen aus.
- CP-CP-Sitzungen zwischen einem Endknoten und einem Netzknoten stellen die folgenden Funktionen bereit:
  - Registrierung von Ressourcen
  - Routing von Verwaltungsdaten (wie Alerts) zwischen dem Endknoten und einem zentralen Alert-Verarbeitungssystem
  - Routing von Topologiedaten jedes Endknotens zum zugehörigen Netzknotenserver. Anhand dieser Daten kann der Netzknotenserver eine Route berechnen, die nicht über den Netzknotenserver führt.
- Über CP-CP-Sitzungen zwischen benachbarten Netzknoten werden Topologieinformationen ausgetauscht. Im Ergebnis dieses Austausches erstellt jeder Netzknoten eine interne Netztopologiedatenbank.

Wenn Sie einen Knoten einrichten, müssen Sie den CP-Namen definieren. Der CP ist gleichzeitig eine LU, die Benutzersitzungen unterstützen kann, und - wenn Sie eine solche Konfiguration wünschen - die einzige definierte LU Ihres Knotens.

## Ressourcen lokalisieren

Im Rahmen der Unterstützung für die Kommunikation zwischen Transaktionsprogrammen baut Communications Server für Linux zunächst eine Sitzung zwischen den logischen Einheiten auf, die diese Transaktionsprogramme steuern. Der Steuerpunkt eines Knotens kann mit APPN LUs im gesamten APPN-Netz lokalisieren, ohne dass der Knoten über Konfigurationsdaten der fernen LU informiert sein muss. Die APPN-Funktion, die LUs im Netz dynamisch lokalisiert, wird als Verzeichnisdienste bezeichnet. Sobald eine Ressource gefunden wurde, wird für die Sitzung eine Route durch das APPN-Netz berechnet.

## Ressourcennamen

Jeder Knoten hat einen eindeutigen Namen, der aus zwei Teilen besteht: einem Netznamen und einem Steuerpunktnamen. Beide Teile zusammen bilden einen vollständig qualifizierten CP-Namen. Anhand dieses Namens kann jeder Knoten von allen anderen Knoten im Netz identifiziert werden. In gleicher Weise ist auch

jede LU durch einen vollständig qualifizierten LU-Namen gekennzeichnet, der sich aus einem Netznamen und einem LU-Namen zusammensetzt.

**Anmerkung:** Weitere Informationen zu Netznamenskonventionen finden Sie in der Veröffentlichung *Einstieg in Communications Server für Linux*.

### Verzeichnisdienste

Jeder APPN-Knoten verwaltet ein Verzeichnis mit Netzressourcen. Die Verzeichnisdienste sind die Komponente des Knotensteuerpunkts, die die lokale Verzeichnisdatenbank verwaltet und (wenn es sich um einen Netzknoten handelt) im gesamten APPN-Netz nach Netzressourcen sucht.

Wenn der Knoten initialisiert ist, nehmen die Verzeichnisdienste die folgenden Informationen auf:

- Art des Knotens (APPN-Netzknoten, APPN-Endknoten oder LEN-Knoten)
- Netz-ID des Knotens
- CP-Name des Knotens

In jedem Knotenverzeichnis werden Einträge für Ressourcen (LUs und PUs) verwaltet, die unter anderem den vollständig qualifizierten Namen, den Typ und den Registrierungsstatus der jeweiligen Ressource enthalten. Welche Ressourcen in den einzelnen lokalen Verzeichnissen gespeichert werden, hängt wie folgt von der Art des Knotens ab:

- Ein LEN-Knoten verwaltet ein Verzeichnis, das seine eigenen LUs enthält. Die Konfiguration muss außerdem Verzeichniseinträge für alle potenziellen Partner-LUs umfassen. Wenn der LEN-Knoten Sitzungen mit LUs des APPN-Netzes aufbauen soll, müssen diese LUs auf dem LEN-Knoten so definiert sein, als würden sie sich auf dem Netzknotenserver des LEN-Knotens befinden. Der LEN-Knoten führt Sitzungen mit den LUs auf seinem Netzknotenserver aus. Der Netzknoten leitet die Sitzung über das APPN-Netz zum entsprechenden Knoten im Netz weiter.

Ein Verzeichniseintrag auf einem LEN-Knoten kann auch Platzhalter enthalten, um mehrere Partner-LUs, die über eine bestimmte Verbindung erreichbar sind, anzugeben.

- Ein APPN-Endknoten verwaltet ein Verzeichnis, das seine eigenen LUs enthält. Der Knoten kann auch so konfiguriert werden, dass er Verzeichniseinträge für Partner-LUs von Nachbarknoten speichert. Auf diese Weise können lokale LUs mit diesen LUs ohne APPN-Funktionen Peer-zu-Peer-Sitzungen aufbauen.

Wenn eine Ressource nicht lokal auf einem Endknoten definiert oder derzeit für den Knoten nicht erreichbar ist, sendet der Endknoten eine Anforderung an seinen Netzknotenserver und fordert diesen auf, das APPN-Netz nach der Ressource zu durchsuchen.

- Ein APPN-Netzknoten verwaltet ein Verzeichnis, das seine eigenen LUs und die LUs der Endknoten und LEN-Knoten seiner Domäne enthält. Ein Endknoten kann seine LUs dynamisch auf seinem Netzknotenserver registrieren lassen. (LEN-Knoten können LUs nicht auf einem Netzknotenserver registrieren. Die LUs von LEN-Knoten müssen deshalb auf den zugehörigen Netzknotenservern konfiguriert werden.) Das Verzeichnis eines Netzknotens kann außerdem zwischengespeicherte Einträge für LUs enthalten, die sich nicht in der Domäne des Netzknotens befinden, deren Position jedoch bei einer ausgeführten Suche festgestellt wurde.

Netzknoten stellen für andere Knoten die beiden folgenden Dienste bereit:

- Suche nach fernen Ressourcen bei entsprechenden Sitzungsanforderungen von Endknoten oder LEN-Knoten
- Positives Beantworten von Suchanforderungen anderer Knoten, wenn eine benannte Ressource im lokalen Verzeichnis gefunden wird.

**LEN-Knotenverzeichnisse:** Abb. 5 zeigt ein Beispiel für ein LEN-Knotenverzeichnis. Da LEN-Knoten keine Unterstützung für CP-CP-Sitzungen bieten, muss das Verzeichnis für den LEN-Knoten LEN1 alle LUs enthalten, mit denen der Knoten kommuniziert. Das Verzeichnis für LEN1 gibt als Position aller LUs, die sich nicht auf einem benachbarten Peerendknoten befinden, den Netzknotenserver (NKA) an. Da der Knoten LEN1 nur über den Knoten NKA auf die LUs zugreifen kann, definiert er den CP des Netzknotens als den „Eigner-CP“ aller LUs, einschließlich der auf den Endknoten befindlichen LUs.

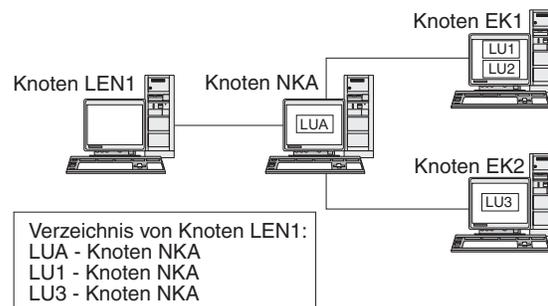


Abbildung 5. LEN-Knotenverzeichnis

Zum Aufbauen einer Sitzung mit einer LU auf einem nicht direkt angeschlossenen Knoten sendet der Knoten LEN1 eine BIND-Anforderung (für die Aktivierung einer LU-LU-Sitzung) an seinen Netzknotenserver (Knoten NKA). Der Server lokalisiert die Ziel-LU automatisch und leitet die BIND-Anforderung weiter.

**Anmerkung:** In diesem Beispiel kann der Knoten LEN1 über seinen Netzknotenserver NKA eine Sitzung mit LU1 auf dem Knoten EK1 aufbauen. LU2 auf dem Knoten EK1 ist im Verzeichnis des Knotens LEN1 nicht definiert, so dass der Knoten LEN1 mit dieser LU keine Sitzung aufbauen kann.

**Endknotenverzeichnisse:** Wenn eine LU nicht in einem Endknotenverzeichnis angegeben ist, leitet der Endknoten mit LOCATE eine Suche nach der gewünschten LU ein. Zum Aktivieren der Suche nach einer fernen LU ruft der Endknoten die Dienste seines Netzknotenservers auf. Abb. 6 auf Seite 20 zeigt ein Beispiel für ein Endknotenverzeichnis.

## APPN-Basiskonzepte

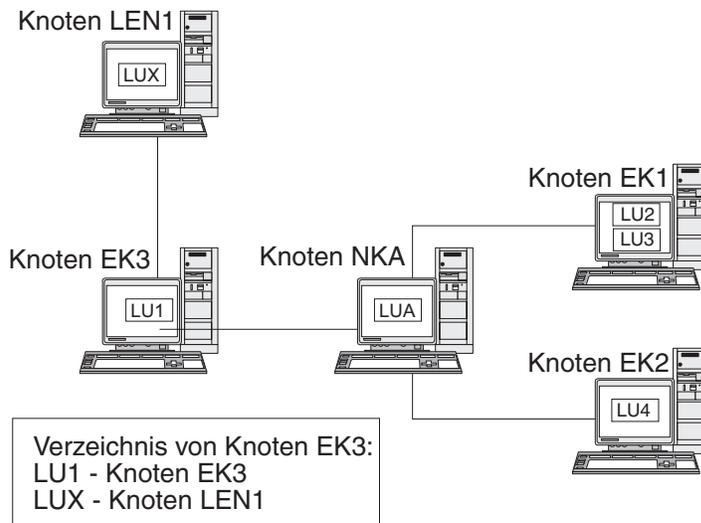


Abbildung 6. Endknotenverzeichnis

Potenzielle Partner-LUs im APPN-Netz müssen nicht auf dem Endknoten definiert sein. Wenn der Knoten EK3 jedoch eine Sitzung mit LUX auf dem Knoten LEN1 aufbauen will, muss die LU des LEN-Knotens auf dem Knoten EK3 als Partner-LU konfiguriert sein.

**Netzknotenverzeichnisse:** Ein Netzknoten stellt für die Endknoten, die er bedient, verteilte Verzeichnisdienste bereit.

Abb. 7 zeigt ein Beispiel für ein Netzknotenverzeichnis.

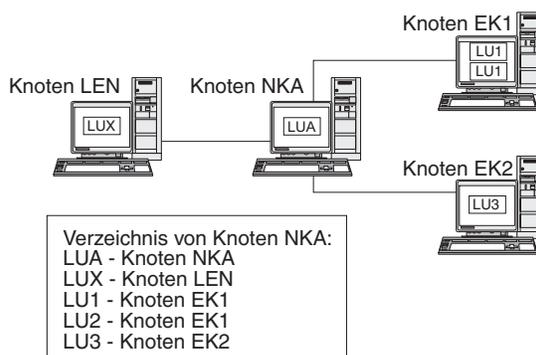


Abbildung 7. Netzknotenverzeichnis

Ein Netzknoten lokalisiert eine ferne LU wie folgt:

1. Der Netzknoten empfängt eine Lokalisierungsanforderung für eine LU. Die Anforderung kann auf eine der folgenden Arten an den Netzknoten übergeben werden:
  - Ein Endknoten oder LEN-Knoten sendet den Namen einer Ziel-LU an seinen Netzknotenserver.
  - In einer LOCATE-Suchanforderung von einem Endknoten ist ein LU-Name angegeben.
  - In einer BIND-Anforderung von einem LEN-Knoten ist ein LU-Name angegeben.
  - Ein Transaktionsprogramm auf dem Netzknoten gibt einen LU-Namen an.

2. Wenn sich die Ziel-LU nicht auf dem Netzknoten befindet, jedoch im Verzeichnis des Netzknotens angegeben ist, sendet der Netzknoten eine gezielte Suchanforderung an den Zielnetzknottenserver, um die Position der LU zu bestätigen. Ist die LU nicht im Netzknotenverzeichnis enthalten, leitet der Knoten eine Durchsuchung des Netzes ein, indem er im Broadcast-Betrieb an jeden Nachbarknoten eine Suchanforderung sendet.
3. Jeder Knoten gibt diese Suchanforderung weiter und meldet Erfolg oder Fehlschlag der Suche zurück.

Ein Netzknoten speichert Informationen, die er durch erfolgreiche Suchaktionen im Broadcast-Betrieb erhalten hat, für seinen künftigen Bedarf im Cache.

Ein APPN-Endknoten kann auch LOCATE-Anforderungen von seinem Netzknotenserver empfangen, durch die er angewiesen wird, nach bestimmten LUs zu suchen oder das Vorhandensein bestimmter LUs zu bestätigen.

Jeder APPN-Endknoten lässt seine LUs auf dem Netzknotenserver registrieren, indem er dem Netzknoten eine Registrierungsnachricht sendet. Auf diese Weise aktualisiert der Netzknoten die Verzeichnisinformationen für die Endknoten seiner Domäne. Ein LEN-Knoten kann keine LUs auf seinem Netzknotenserver registrieren. Deshalb müssen alle LUs des LEN-Knotens auf dem Netzknotenserver konfiguriert werden.

## Sitzungs-Routing

APPN unterstützt die folgenden Prozeduren für eine dynamische Routenauswahl:

- Für Sitzungen mit Nachbarknoten: direktes Sitzungs-Routing.
- Für Sitzungen, die über einen oder mehrere Zwischenknoten weitergeleitet werden, eine der folgenden Prozeduren:
  - Weitervermittlung einer Sitzungsroute (ISR, Intermediate Session Routing). Bei dieser Prozedur wird eine Route bereitgestellt, die sich im Verlaufe der Sitzung nicht ändert.
  - High-Performance Routing (HPR). Zu dieser Prozedur gehören das Rapid Transport Protocol (RTP) und ANR-Funktionen (Automatic Network Routing). Mit RTP können Sie den Zyklus- und Speicherbedarf für das Routing von Paketen der Vermittlungsschicht über Zwischenknoten auf einer Sitzungsroute minimieren und mit ANR den Sitzungsverkehr bei Überlastung oder Ausfall einer Route umleiten.

Die APPN-Funktionen für dynamische Routenauswahl werden als Topologie- und Routing-Dienste (TRS, Topology and Routing Services) bezeichnet.

## Topologie- und Routing-Dienste

Jeder APPN-Knoten verfügt über eine Topologiedatenbank, in der Informationen zu anderen APPN-Knoten und zu Verbindungsgruppen gespeichert werden. Verbindungsgruppen sind Gruppen von Verbindungen zwischen zwei bestimmten Knoten. Der Inhalt der Datenbank eines Knotens hängt von der Art des Knotens ab:

- Alle Netzknoten verfügen über eine Kopie der Netztopologiedatenbank. Diese gemeinsame Datenbank enthält Informationen zu allen anderen Netzknoten, z. B. Netz-IDs, CP-Namen und andere Knotenkenndaten, und zu den Verbindungsgruppen zwischen den einzelnen Knotenpaaren. Diese Datenbank gibt einen vollständigen Überblick über die Topologie der Zentralverbindung des Netzes, d. h. über die Knoten und Verbindungsgruppen, die für das Routing von Sitzungen zwischen allen Knotenpaaren im Netz verwendet werden können.

## APPN-Basiskonzepte

Die Topologiedatenbank auf den einzelnen Netzknoten enthält aber auch Informationen zu Verbindungsgruppen vom jeweiligen Knoten zu benachbarten Endknoten oder LEN-Knoten.

Der Netzknoten berechnet mit Hilfe der Daten in der Topologiedatenbank Routen für Sitzungen zwischen LUs seiner Domäne und fernen LUs oder stellt anderen Netzknoten Daten zur Verfügung, damit diese Sitzungsrouten berechnen können.

- Jeder Endknoten verfügt über eine lokale Topologiedatenbank mit Informationen zu Verbindungsgruppen von diesem Endknoten zu den Nachbarknoten.

Der Endknoten stellt diese Informationen dem Netzknotenserver zur Verfügung, wenn er bei diesem die Suche nach einer LU oder die Berechnung einer Sitzungsrouten zu dieser LU anfordert. Der Netzknotenserver greift bei der Berechnung der Sitzungsrouten für den Endknoten auf diese Topologiedaten des Endknotens zurück. Der Endknoten nutzt diese Informationen, wenn er Sitzungen mit vorab definierten LUs von Nachbarknoten aufbaut. Die Topologiedatenbank des Endknotens unterstützt nur die Kommunikation mit Nachbarknoten.

### Anmerkung:

1. APPN-Netzknoten und -Endknoten speichern auch Topologieinformationen über Verbindungen zu einem Verbindungsnetz (siehe Abschnitt „APPN-Verbindungsnetze“ auf Seite 27).
2. LEN-Knoten speichern lokale Topologiedaten. Sie leiten diese Informationen nicht an einen Netzknotenserver weiter.

Wie in Abb. 8 auf Seite 23 gezeigt, werden die Netztopologiedaten auf allen Netzknoten nachgebildet und die lokalen Topologiedaten auf Netzknoten und Endknoten gespeichert.

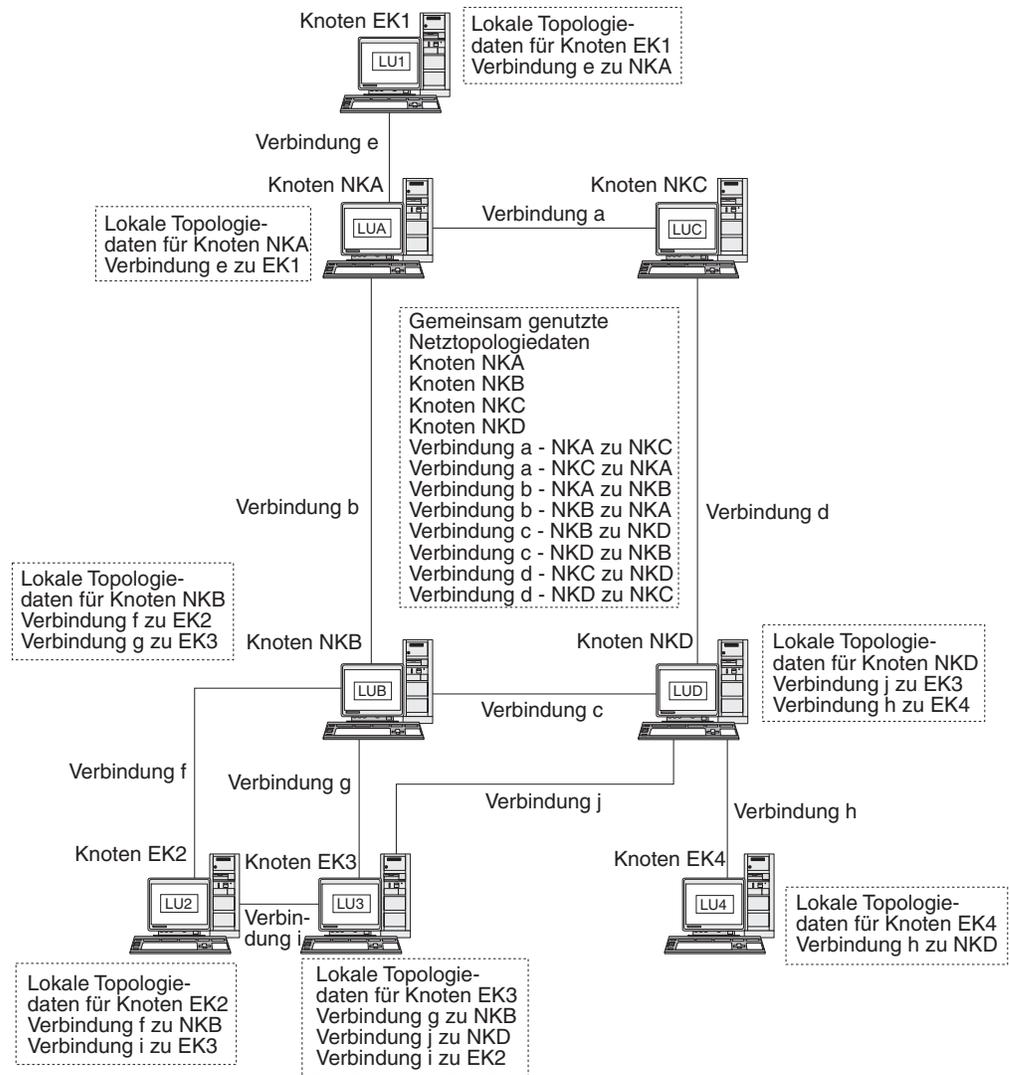


Abbildung 8. Netztopologiedatenbank auf Netzknotten

Die gemeinsame Topologiedatenbank wird auf den Knoten NKA, NKB, NKC und NKD dupliziert. Jeder dieser Knoten speichert außerdem lokale Topologiedaten (mit Ausnahme des Knotens NKC, der nicht über Verbindungen zu Endknotten und deshalb auch nicht über lokale Topologieinformationen verfügt). Der Knoten NKB hat beispielsweise Informationen über die Verbindung 'f' zum Knoten EK2 und die Verbindung 'g' zum Knoten EK3. Er hat jedoch keine Informationen über die Verbindung 'i', die die Knoten EK2 und EK3 verbindet.

Endknotten speichern nur Informationen über Verbindungen zu Nachbarknoten. Der Knoten EK2 hat beispielsweise Informationen über die Verbindung 'f' zum Knoten NKB und die Verbindung 'i' zum Knoten EK3.

**Topologiedatenbank aktualisieren:** APPN-Netzknotten tauschen über CP-CP-Sitzungen Netztopologiedaten aus, wenn eine Ressource (z. B. ein Knoten oder eine Verbindung zwischen zwei Netzknotten) aktiviert bzw. inaktiviert wird oder sich die Kenndaten einer vorhandenen Ressource ändern. Nach einer solchen Änderung generiert der Netzknotten eine Aktualisierung der Topologiedatenbank mit Knotenidentifikation, Knoten- und Verbindungskenndaten und Aktualisierungsfolge-

nummern, die die zu aktualisierende Ressource und die Änderungen für diese Ressource angeben. Jede Aktualisierung der Topologiedatenbank wird an alle aktiven Netzknoten gesendet, um sicherzustellen, dass die Netztopologiedatenbank im gesamten Netz auf dem aktuellen Stand ist.

**Routenauswahl in einem APPN-Netz:** Die APPN-Verzeichnisdienste suchen einen bestimmten Sitzungspartner. Nachdem der Sitzungspartner im Netz gefunden wurde, berechnen die Topologie- und Routing-Dienste die optimale Sitzungsrouten. Jeder Netzknoten stellt Routenauswahldienste für Sitzungen bereit, die von seinen eigenen LUs und von LUs der von ihm bedienten Endknoten oder LEN-Knoten eingeleitet werden. Für die dynamische Berechnung von Routen zwischen Knoten nutzt ein Netzknoten seine eigenen Topologiedaten und die Informationen der gemeinsamen Netztopologiedatenbank.

Nach der Lokalisierung des Sitzungspartners führt der Netzknoten die folgenden Schritte aus, um eine Route auszuwählen:

1. Der Netzknoten stellt die erforderlichen Kenndaten für die Sitzungsrouten zusammen.  
Die LU, die die Sitzung anfordert, gibt einen Modusnamen an, der die Sitzungskenndaten bestimmt. Der zugeordnete Modus bezeichnet eine Serviceklasse, die Voraussetzungen für die zur Weiterleitung des Sitzungsverkehrs verwendeten Verbindungen festlegt.
2. Der Netzknoten stellt wie folgt Informationen zu allen Verbindungsgruppen und Netzknoten für potenzielle Routen zusammen:
  - Wenn die Sitzung von einem Endknoten angefordert wurde, stellt der Endknoten Informationen zu Verbindungen bereit, die er zu seinem Netzknotenserver und ggf. zu einem Verbindungsnetz hat.
  - Ist der Sitzungspartner kein Nachbarknoten, identifiziert der Netzknotenserver für die die Sitzung anfordernde LU mit Hilfe der Netztopologiedatenbank Netzknoten und Zwischenverbindungsgruppen für die Route zum Sitzungspartner.
  - Befindet sich der Sitzungspartner auf einem Endknoten, stellt der Endknoten (oder sein Netzknotenserver) Informationen zur Verbindung zwischen dem Netzknotenserver und dem Endknoten (oder zur Verbindung zwischen dem Endknoten und einem Verbindungsnetz) bereit.
3. Der Netzknoten schließt alle Netzknoten und Verbindungsgruppen aus, die die für die Sitzungsrouten angegebenen Voraussetzungen nicht erfüllen.
4. Der Netzknoten berechnet die für die Sitzung optimale Route.

Der Algorithmus für die Routenberechnung berechnet in Abhängigkeit von der angegebenen Serviceklasse für jeden Knoten und jede logische Verbindung die Wertigkeit. Die so berechneten Werte werden zu einer gewichteten Summe für jede Route addiert. Für die Auswahl des optimalen Pfades errechnet der Netzknoten die Route mit der derzeit niedrigsten Wertigkeit vom Knoten mit der Ausgangs-LU zum Knoten mit der Ziel-LU.

### Zwischenpfadsteuerung

Mit Hilfe dieser Funktion kann ein APPN-Netzknoten für einen anderen Knoten bestimmte Daten empfangen und weiterleiten. Ursprung und Ziel der Daten kann ein Endknoten, ein anderer Netzknoten oder ein LEN-Knoten sein.

Die Zwischenpfadsteuerung unterstützt Sitzungen zwischen LUs, die sich nicht auf benachbarten Knoten befinden. Nachdem für eine Sitzung eine Route ausgewählt wurde, leiten die auf dieser Route liegenden APPN-Netzknoten die Sitzungsdaten über die Zwischenpfadsteuerung an den nächsten auf der Route liegenden Knoten weiter.

Zu den in der Topologiedatenbank gespeicherten Ressourcenkenndaten kann auch der Überlastungsstatus gehören. Wenn ein Netzknoten stark überlastet ist, kann der Knoten diese Information an andere Knoten im Netz weitergeben. Dadurch sinkt die Wahrscheinlichkeit, dass der überlastete Knoten in Sitzungsrouten aufgenommen wird, die für neue Sitzungen berechnet werden.

APPN stellt zwei Arten der Zwischenpfadsteuerung bereit:

- Bei der Weitervermittlung einer Sitzungsrouten (ISR, Intermediate Session Routing), die auf allen Netzknoten verfügbar ist, protokolliert der Netzknoten jede einzelne Zwischensitzung. Jeder Transitknoten reguliert die Dosierung der Sitzungsdaten und steuert so die Geschwindigkeit, mit der die Daten zwischen den benachbarten Knoten fließen. Darüber hinaus ist jeder Transitknoten in der Lage, Daten zu segmentieren und segmentierte Daten wieder zusammzusetzen. Typisch für die Weitervermittlung einer Sitzungsrouten ist, dass eine einmal eingerichtete Sitzungsrouten von allen Sitzungsdaten verwendet wird. Fällt ein Teil der Routen aus, wird die Sitzung beendet.
- Beim Automatic Network Routing (ANR), das auf allen die HPR-Funktion von APPN unterstützenden Netzknoten verfügbar ist, können Transitnetzknoten den Sitzungsverkehr dynamisch umleiten, falls ein Teil der Sitzungsrouten ausfällt. ANR erlaubt keine Nachrichtendosierung für Zwischensitzungen und keine Segmentierung von Daten oder Zusammensetzung segmentierter Daten.

Mit ANR können Transitknoten den Sitzungsverkehr erheblich schneller weiterleiten, als dies mit der traditionellen APPN-Funktion ISR möglich ist. Die ANR-Funktion erfordert jedoch an den RTP-Endpunkten (Rapid Transport Protocol) einen erhöhten Systemaufwand. Erstreckt sich eine Routen über nur wenige Transitknoten, kann die Übertragung auf einer ANR-Routen aufgrund der Verarbeitungszeit an den Endpunkten sogar langsamer als die Übertragung auf einer ISR-Routen sein. Bei Routen mit einer größeren Zahl von Transitknoten (Zwischenschritten) kann mit ANR in der Regel eine höhere Geschwindigkeit erreicht werden. Bei welcher Anzahl von Transitknoten der Durchsatz für beide Funktionen identisch ist, hängt von der Effektivität der RTP-Knoten ab.

### Direkte Konnektivität

Direkte Konnektivität bedeutet, dass der Sitzungsverkehr direkt und ohne Sitzungsrouten durch einen APPN-Netzknoten zwischen zwei Knoten transportiert werden kann. Sitzungen zwischen direkt verbundenen Knoten ermöglichen in der Regel einen schnelleren Datenaustausch als Sitzungen, bei denen die Daten über ein Netz weitergeleitet werden. Für Knoten in einem SATF-Netz (Shared-Access Transport Facility) könnte die Effektivität erhöht werden, indem für jedes Knotenpaar im Netz Verbindungen definiert werden. Dies gilt beispielsweise für die in Abb. 9 auf Seite 26 gezeigten Knoten auf einem Token-Ring. Dies kann jedoch eine schwierige Aufgabe sein, denn die Anzahl der Verbindungsstationen ist gleich  $n \times (n-1)$ , wobei  $n$  die Zahl der Knoten im Netz repräsentiert.

Abb. 9 auf Seite 26 zeigt ein APPN-Netz auf einem Token-Ring.

## APPN-Basiskonzepte

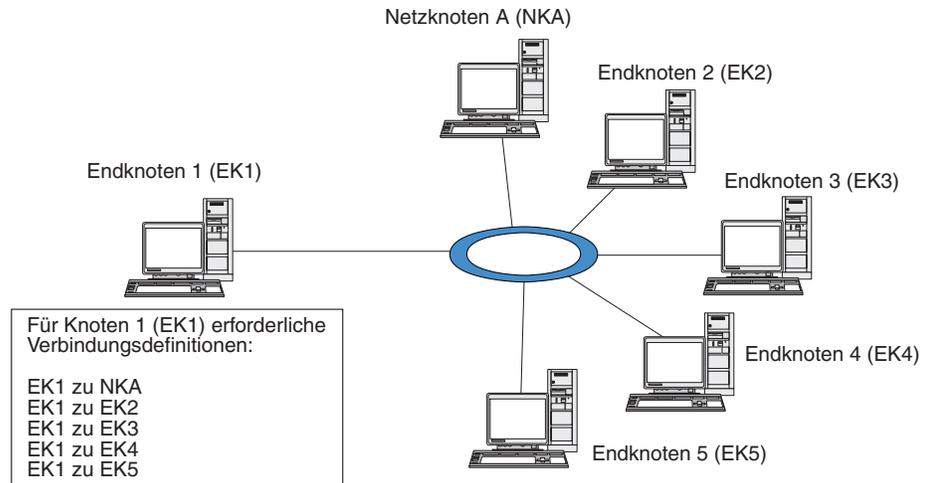


Abbildung 9. APPN-Netz mit SATF-Einrichtung

Wenn der Knoten EK1 für jede Verbindung im Netz eine Definition gespeichert hat, kann er zu jedem beliebigen Knoten eine direkte Verbindung herstellen. Abb. 10 veranschaulicht die Verbindungsdefinitionen, die zur Unterstützung direkter Verbindungen zwischen dem Knoten EK1 und allen anderen Knoten im APPN-Netz erforderlich sind. Wenn das Netz fünf weitere Knoten enthält, benötigt der Knoten EK1 die folgenden fünf Verbindungsdefinitionen:

- EK1 zu NKA
- EK1 zu EK2
- EK1 zu EK3
- EK1 zu EK4
- EK1 zu EK5

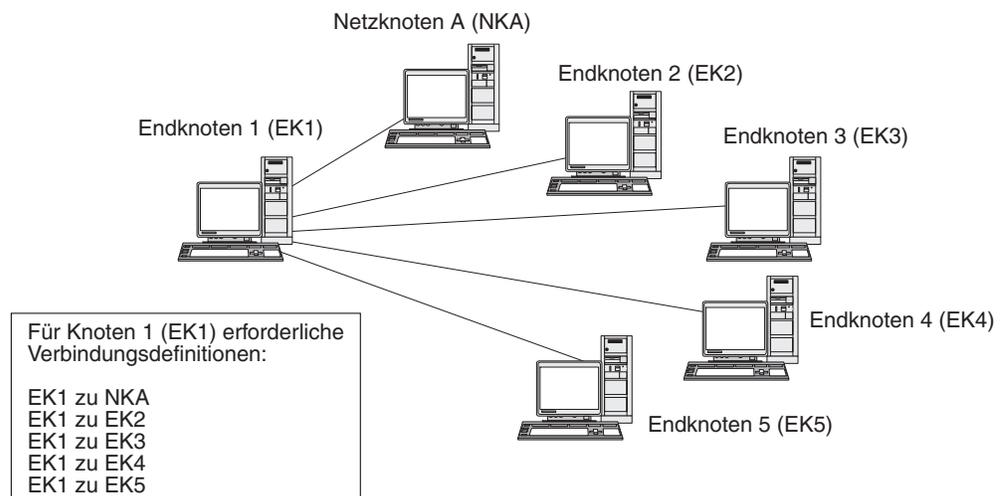


Abbildung 10. Für direkte Verbindungen zwischen dem Knoten EK1 und allen anderen Knoten eines APPN-Netzes erforderliche Definitionen

Sollen alle Knoten im Netz direkte Verbindungen zu jedem anderen Knoten unterstützen, wären für die sechs Knoten dieses Beispielnetzes insgesamt 30 Verbindungsdefinitionen notwendig. Die Zahl der Verbindungsdefinitionen kann nach der Formel  $n \times (n-1)$  berechnet werden, wobei  $n$  die Zahl der Knoten im Netz ist. In einem größeren Netz wird die Anzahl der Verbindungsdefinitionen schnell unüberschaubar. Eine steigende Zahl von Verbindungsdefinitionen zieht eine wachsende Zahl von Aktualisierungen der Topologiedatenbank nach sich, wodurch die Netzleistung herabgesetzt werden kann.

APPN-Verbindungsnetze bieten eine Lösung für dieses Problem.

### APPN-Verbindungsnetze

Für APPN-Netze, die an eine SATF-Einrichtung (Shared-Access Transport Facility) angeschlossen sind, wird die Zahl der für die Unterstützung der direkten Konnektivität zwischen Knoten im Netz erforderlichen Verbindungsdefinitionen durch ein APPN-Verbindungsnetz stark reduziert. In einem Verbindungsnetz muss ein APPN-Endknoten anstelle von Verbindungen zu allen Knoten im Netz nur eine Verbindung zu einem benachbarten Netzknotenserver und eine Verbindung zum Verbindungsnetz konfigurieren.

Das Verbindungsnetzfeature kann von APPN-Netzen genutzt werden, die folgende Bedingungen erfüllen:

- Die Knoten im APPN-Netz müssen über Übertragungsmedien mit Wählleitungen wie Token-Ring oder Ethernet verbunden sein.
- Alle Verbindungen im APPN-Verbindungsnetz müssen dieselben Übertragungsmedien nutzen.
- Das APPN-Netz, das das Verbindungsnetz enthält, muss vollständig verbunden sein. In einem vollständig verbundenen Netz hat jeder Knoten mindestens eine Verbindung zu einem Nachbarknoten, die CP-CP-Sitzungen unterstützt.

In einem Verbindungsnetz wird die SATF-Einrichtung als virtueller Routing-Knoten (VRN, Virtual Routing Node) genutzt, der direkt an die einzelnen Knoten im Verbindungsnetz angeschlossen wird. Der Name des Verbindungsnetzes wird für den VRN als Steuerpunktnamen verwendet. Der VRN unterstützt das direkte Routing von Sitzungsdaten zwischen zwei beliebigen Knoten im Verbindungsnetz, baut jedoch keine CP-CP-Sitzungen mit anderen Knoten auf und generiert keine Aktualisierungen der Topologiedatenbank. Für jeden Knoten im Verbindungsnetz ist nur eine Verbindung zum zugehörigen Netzknotenserver erforderlich.

Abb. 11 auf Seite 28 zeigt die bei Verwendung eines Verbindungsnetzes benötigten Verbindungsdefinitionen. Mit Hilfe eines virtuellen Knotens unterstützt das Verbindungsnetz direkte Verbindungen zwischen dem Knoten EK1 und jedem anderen Knoten im APPN-Netz. Es sind somit nur zwei Verbindungsdefinitionen erforderlich.

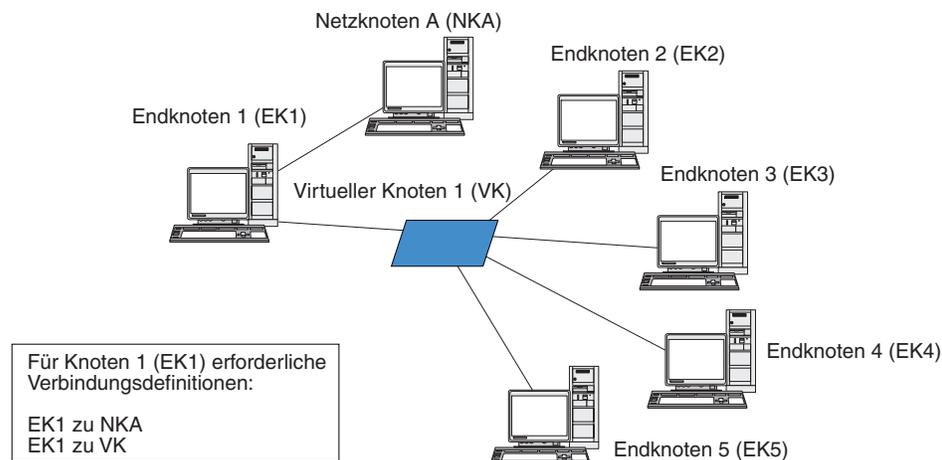


Abbildung 11. Für direkte Verbindungen über einen virtuellen Knoten benötigte Definitionen

Für die Unterstützung direkter Verbindungen zwischen allen Endknotenpaaren im APPN-Netz wären insgesamt zehn Verbindungsdefinitionen erforderlich. (Jeder Endknoten benötigt zwei Verbindungsdefinitionen: eine für die Verbindung zu einem Netznotenserver und eine für die Verbindung zum virtuellen Knoten.) Verglichen mit den Anforderungen für die direkte Konnektivität an ein APPN-Netz ohne Verbindungsnetz (siehe Abb. 10 auf Seite 26) ergibt sich eine deutlich geringere Zahl von Verbindungsdefinitionen (in diesem Beispiel 10 anstelle von 30). In einem größeren Netz fällt der Unterschied bei den erforderlichen Definitionen erheblich größer aus.

Im Verbindungsnetz wird eine Sitzung zwischen den LUs zweier Knoten wie folgt aufgebaut:

1. Zunächst baut jeder Endknoten eine CP-CP-Sitzung mit seinem Netznotenserver auf. (Wenn zwei Endknoten verschiedene Netznotenserver haben, müssen diese Netznoten über eine Verbindung verfügen, die CP-CP-Sitzungen unterstützt.)
2. Die Endknoten melden dem Netznotenserver ihre VRN-Verbindungen und lokalen Adressen. Die angegebene lokale Adresse kann die Adresse eines Servicezugriffspunkts (SAP) und eine MAC-Adresse sein.
3. Der Server wählt als optimale Route für die LU-LU-Sitzung normalerweise die direkte Verbindung zwischen zwei Endknoten aus. Er stellt für den Knoten mit der primären LU Informationen bereit, die dieser für das Herstellen einer dynamischen Verbindung zum Knoten mit der Partner-LU benötigt.
4. Anschließend können die Knoten ohne Weitervermittlung einer Sitzungsroute eine LU-LU-Sitzung aufbauen.

### Branch Extender

Wie bereits in den vorherigen Abschnitten erwähnt, müssen Netznoten in einem APPN-Netz Topologiedaten (zur Position anderer Knoten im Netz und zu den zwischen diesen bestehenden DFV-Verbindungen) verwalten und diese Daten im Netz weiterleiten, sobald sich die Topologie ändert. Wächst das Netz, nehmen auch die Anzahl der gespeicherten Daten und der topologiebezogene Datenaustausch im Netz zu. Dies erschwert die Verwaltung der Daten.

Diese Probleme können umgangen werden, indem das Netz in Teilnetze gegliedert wird. Jeder Endknoten muss dann nur die Topologiedaten der Knoten in seinem

eigenen Teilnetz verwalten. Der Versuch, Ressourcen in anderen Teilnetzen zu lokalisieren, führt dann allerdings zu einem vermehrten Datenaustausch im Netz.

Das in Abb. 12 dargestellte APPN-Feature Branch Extender bietet eine Lösung für diese Probleme an.

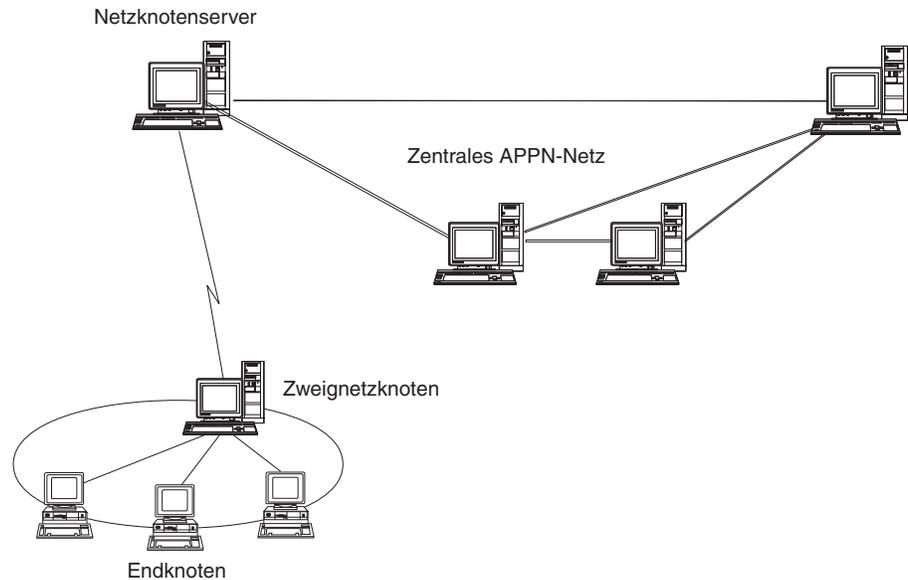


Abbildung 12. Branch Extender

Wie der Name bereits nahe legt, wurde Branch Extender für Netze entwickelt, die in bestimmte Bereiche, z. B. die verschiedenen Zweigstellen einer großen Organisation, unterteilt werden können. Branch Extender gliedert Zweignetze aus dem zentralen APPN-Netz (z. B. dem Netz am Hauptsitz der Organisation) aus.

Jedes Zweignetz enthält einen neuen Knotentyp, den Zweignetzknoten, der mit einem Netzknottenserver im zentralen APPN-Netz verbunden ist. Der Zweignetzknoten vereint in sich die Funktionen eines APPN-Netzknottenservers und eines APPN-Endknotens.

- Aus Sicht des zentralen Netzes ist der Zweignetzknoten ein Endknoten, der mit dem zugehörigen Netzknottenserver im zentralen Netz verbunden ist.
  - Die Knoten im zentralen Netz haben keine Kenntnis von den Knoten innerhalb des Zweignetzes, so dass weniger Topologiedaten gespeichert werden müssen.
  - Weil der Zweignetzknoten als Endknoten erscheint, empfängt er keine Topologiedaten vom zentralen Netz. (Topologiedaten werden nur zwischen Netzknottenservern übertragen.)
  - Der Zweignetzknoten registriert alle Ressourcen im Zweignetz auf seinem Netzknottenserver, als würden sie sich auf dem Zweignetzknoten selbst befinden. Die Knoten im zentralen Netz können somit Ressourcen im Zweignetz suchen, ohne Kenntnis davon zu haben, dass zu diesem Zweignetz verschiedene Knoten gehören.
- Aus Sicht des Zweignetzes ist der Zweignetzknoten ein Netzknottenserver, der für Endknoten im Zweignetz die Funktion eines Netzknottenservers übernimmt. Für alle Knoten im Zweignetz stellt sich ihre Verbindung zum übrigen Netz so dar, als wäre sie eine Verbindung über ihren Netzknottenserver, so wie es bei Standardnetzknottenservern der Fall ist.

### Von APPN-Netzen auf Unterbereichsnetze zugreifen

In APPN-Netzen müssen die Netzressourcen nicht von einem Host gesteuert werden. Dennoch enthalten viele APPN-Netze einen Host. APPN wurde bereits auf vielen Hostplattformen implementiert und gestattet den Hosts, im APPN-Netz die Rolle von Netzknoten zu übernehmen und im SNA-Unterbereichsnetz weiterhin einen SSCP für die Steuerung der SNA-Funktionen bereitzustellen.

Viele SNA-Netze enthalten Elemente eines SNA-Unterbereichsnetzes und eines APPN-Netzes. Die Zentralverbindung des Netzes wird von Netzknoten gebildet, die die Lücke zwischen einer abhängigen LU und den Einrichtungen des Hosts überbrücken müssen. Hierfür sind zwei zusätzliche Dienste notwendig:

- Die DLUS-Funktion (Dependent LU Server) auf dem Host ermöglicht den Zugriff auf die alten SSCP-Funktionen und bildet die Schnittstelle zum APPN-Netz.
- Mit Hilfe der DLUR-Funktion (Dependent LU Requester) auf einem Netzknoten oder Endknoten kann der Sitzungsverkehr von abhängigen LUs über ein APPN-Netz zu einem Host transportiert werden. Über DLUR können abhängige LU-Sitzungen von den flexibleren APPN-Routing-Funktionen profitieren.

Mit Hilfe der Kombination aus DLUR und DLUS (die in der Regel kurz als DLUR bezeichnet wird) kann abhängiger LU-Verkehr über die APPN-Zentralverbindung transportiert werden. Vorhandene SNA-Anwendungen, die mit abhängigen LUs arbeiten, können ohne Änderung weiterhin verwendet werden und zusätzlich von den Vorzügen der APPN-Netzverwaltung, der dynamischen Ressourcenlokalisierung und dem Dienst für Routenauswahl profitieren. DLUR bietet auf diese Weise einen nützlichen Migrationspfad vom SNA-Unterbereichsnetz zum APPN-Netz.

Die abhängigen LUs müssen sich nicht auf dem Knoten mit der DLUR-Funktion befinden. Wenn die DLUR-Funktion von einem Netzknoten bereitgestellt wird, kann sich die abhängige LU auf einem benachbarten Netzknoten, einem Endknoten oder einem LEN-Knoten befinden. Wird die DLUR-Funktion von einem Endknoten bereitgestellt, muss sich die abhängige LU allerdings auf diesem Endknoten befinden.

---

## Kapitel 2. Communications Server für Linux verwalten

Eine Übersicht über die Verwaltung von Communications Server für Linux und die verschiedenen zur Verfügung stehenden Verwaltungstools finden Sie im Abschnitt „Verwaltung von Communications Server für Linux im Überblick“.

Der erste Schritt bei der Verwaltung von Communications Server für Linux ist das Konfigurieren des Knotens und seiner Ressourcen. Beginnen Sie zunächst mit der Planung der Konfiguration. Gehen Sie dabei wie im Abschnitt „CS-Linux-Konfiguration planen“ auf Seite 38 beschrieben vor.

Bevor Sie Communications Server für Linux konfigurieren können, müssen Sie Communications Server für Linux wie im Abschnitt „Communications Server für Linux auf dem lokalen System aktivieren und inaktivieren“ auf Seite 40 beschrieben aktivieren.

Nach Aktivierung von Communications Server für Linux können Sie das Motif-Verwaltungsprogramm ausführen (siehe Abschnitt „Motif-Verwaltungsprogramm verwenden“ auf Seite 43). Das Motif-Verwaltungsprogramm führt Sie durch die Schritte, die zum Konfigurieren von Communications Server für Linux für die Unterstützung der SNA-Kommunikation erforderlich sind. Das Motif-Verwaltungsprogramm ist das empfohlene Verwaltungstool, denn es reduziert die von Ihnen bereitzustellenden Konfigurationsdaten auf ein Minimum und bietet für jeden Schritt, den Sie für verschiedene Kommunikationsarten (wie 3270- oder APPC-Kommunikation) ausführen müssen, eine Anleitung.

Alternativ dazu können Sie das Befehlszeilenverwaltungsprogramm verwenden. Eine diesbezügliche Beschreibung finden Sie im Abschnitt „Befehlszeilenverwaltungsprogramm verwenden“ auf Seite 55.

Das vorliegende Handbuch enthält zu allen Verwaltungsaufgaben Informationen, die Sie sowohl für die Verwaltung mit dem Motif- als auch dem Befehlszeilenverwaltungsprogramm verwenden können. Im Abschnitt „Verwaltungstools“ auf Seite 32 finden Sie darüber hinaus eine Beschreibung weiterer Konfigurationsmethoden.

---

### Verwaltung von Communications Server für Linux im Überblick

Als Administrator für Communications Server für Linux sind Sie für die Installation der Software und die Verwaltung der CS-Linux-Ressourcen verantwortlich. Bevor Sie mit der Verwaltung von Communications Server für Linux beginnen, müssen Sie die wichtigsten Features von Communications Server für Linux kennen (siehe *Einstieg in Communications Server für Linux*). In diesem Abschnitt sind die von Ihnen auszuführenden Verwaltungsaufgaben und die dafür verfügbaren Tools erläutert.

#### Verwaltungszuständigkeit

Die Verwaltung eines CS-Linux-Systems umfasst die folgenden Schritte:

1. Definieren der Ressourcen des CS-Linux-Systems gemäß den Anforderungen der Benutzerprogramme, die ausgeführt werden sollen. Stellen Sie in Zusammenarbeit mit den Administratoren der Hosts und Peercomputer, mit denen

## Verwaltung von Communications Server für Linux im Überblick

Communications Server für Linux kommuniziert, sicher, dass die Konfiguration von Communications Server für Linux an die des fernen Systems angeglichen ist.

2. Initialisieren Sie die Software Communications Server für Linux.
3. Bei Bedarf steht es Ihnen frei, die Konfiguration dynamisch durch das Hinzufügen oder Entfernen von Ressourcen oder durch das Aktivieren bzw. Inaktivieren definierter Ressourcen zu ändern.
4. Überwachung des Status der aktiven Ressourcen und Abrufen von Diagnoseinformationen zur Bestimmung auftretender Fehler.
5. Bei Bedarf das Erstellen von Anwendungsprogrammen oder Shell-Prozeduren zur Automatisierung von Standardverwaltungsoperationen.

Diese Aufgaben werden normalerweise von einem Systemadministrator ausgeführt, der sich am Standort des CS-Linux-Systems befindet. Communications Server für Linux stellt jedoch die Servicepunkt-Befehlsfunktion (SPCF, Service Point Command Facility) bereit, die es einem Bediener ermöglicht, die Schritte 2 und 3 fern über das Programm NetView auszuführen, indem er Verwaltungsbefehle auf der NetView-Konsole eingibt. Weitere Informationen zu SPCF finden Sie in Kapitel 9, „Communications Server für Linux mit NetView verwalten“, auf Seite 127.

## Verwaltungstools

Communications Server für Linux stellt eine Reihe von Tools für die Verwaltung des Systems bereit. Je nach Anforderung werden Sie vielleicht nicht alle verfügbaren Tools verwenden. Dieser Abschnitt gibt einen Überblick über die Funktionen der einzelnen Tools.

### Anmerkung:

1. Das vorliegende Handbuch enthält allgemeine Informationen zur Verwaltung von Communications Server für Linux. Sie können die Verwaltungsaufgaben mit allen in diesem Abschnitt beschriebenen Tools ausführen. Für die meisten Aufgaben wird das Motif-Verwaltungsprogramm empfohlen, weil es kontextgebundene Anleitungen für die Knotenkonfiguration und -verwaltung bietet.
2. Informationen zur Steuerung der Verwendung der Verwaltungstools von Communications Server für Linux sowie zu den verfügbaren Verwaltungsfunktionen finden Sie im Abschnitt „Berechtigungen für die Verwaltung“ auf Seite 38.

Zu Communications Server für Linux gehören die folgenden Verwaltungstools:

- Motif-Verwaltungsprogramm (siehe Abschnitt „Motif-Verwaltungsprogramm“ auf Seite 33)
- Befehlszeilenverwaltungsprogramm (siehe Abschnitt „Befehlszeilenverwaltungsprogramm“ auf Seite 34) (Hinweise zu diesem Programm finden Sie auch in der Veröffentlichung *Communications Server for Linux Administration Command Reference*.)
- Servicepunkt-Befehlsfunktion (siehe Abschnitt „Funktion für ferne Befehle“ auf Seite 35)
- Konfigurationsdateien (siehe Abschnitt „Konfigurationsdateien“ auf Seite 35)
- Diagnosetools (siehe Abschnitt „Diagnosetools“ auf Seite 37)

Alle Verwaltungstools von Communications Server für Linux verwenden die API NOF. Mit dieser API können Sie auch eigene Verwaltungstools schreiben. Weitere Informationen hierzu finden Sie im Abschnitt „NOF-Anwendungen“ auf Seite 36.

### Motif-Verwaltungsprogramm

Der einfachste Weg, die CS-Linux-Konfiguration zu definieren und zu ändern, ist die Verwendung des Motif-Verwaltungsprogramms (**xsnaadmin**). Dieses Programm stellt eine grafische Benutzerschnittstelle bereit, über die Sie die CS-Linux-Ressourcen anzeigen und verwalten können.

Das Programm bietet die folgenden Verwaltungsoperationen an:

- Definieren von CS-Linux-Ressourcen
- Starten und Stoppen eines Knotens und seiner Konnektivitätsressourcen
- Ändern der Konfiguration definierter Ressourcen
- Abfragen der Konfiguration definierter Ressourcen und, falls die Ressourcen aktiv sind, Abfragen ihres aktuellen Status
- Löschen von Ressourcen

Mit dem Motif-Verwaltungsprogramm können Sie sowohl die Knotenressourcen (für jeden Server im LAN, sofern er die Software Communications Server für Linux ausführt) als auch die Domänenressourcen verwalten. Das Programm führt Sie mit Menüs durch die Konfiguration der notwendigen Ressourcen. Es gibt Menüs für jede Übertragungsart (z. B. 3270 oder APPC).

**Anmerkung:** In Abhängigkeit davon, welche Option Sie in einer bestimmten Dialoganzeige ausgewählt haben, können sich die Fenster und Dialoganzeigen des Motif-Verwaltungsprogramms von den in diesem Handbuch abgebildeten unterscheiden.

Das Motif-Verwaltungsprogramm stellt Hilfeanzeigen mit übersichtlichen Informationen zu SNA und Communications Server für Linux, Referenzinformationen zu CS-Linux-Dialoganzeigen und Anleitungen zur Ausführung bestimmter Aufgaben bereit.

Vergewissern Sie sich vor dem Starten des Motif-Verwaltungsprogramms, dass Communications Server für Linux aktiviert ist (weitere Informationen hierzu finden Sie in Kapitel 2, „Communications Server für Linux verwalten“, auf Seite 31). Wie bei allen X/Motif-Anwendungen müssen Sie gegebenenfalls die Umgebungsvariable DISPLAY definieren, um einen geeigneten X-Server anzugeben.

Wenn Sie das Motif-Verwaltungsprogramm im Hintergrund starten wollen, setzen Sie den folgenden Befehl ab:

```
xsnaadmin &
```

In der Hauptanzeige werden alle gestarteten CS-Linux-Server angezeigt. Für die bereits konfigurierten Server können Sie im Programm einen Knoten auswählen. Zu diesem Knoten wird dann die Konfiguration angezeigt. Sind noch keine Server konfiguriert, fordert das Programm Sie zur Auswahl eines Knotens auf und führt Sie durch die erforderlichen Schritte für das Definieren des Knotens.

Weitere Informationen zur Verwendung des Motif-Verwaltungsprogramms sowie zum Definieren und Verwalten der CS-Linux-Ressourcen können Sie dem Abschnitt „Motif-Verwaltungsprogramm aufrufen“ auf Seite 43 oder den vom Programm bereitgestellten Hilfeanzeigen entnehmen.

**Anmerkung:** Mit dem Motif-Verwaltungsprogramm können Sie alle erforderlichen Parameter für Standardkonfigurationen von Communications Server für Linux definieren. Für erweiterte Parameter gibt das Motif-

## Verwaltung von Communications Server für Linux im Überblick

Verwaltungsprogramm Standardwerte vor. Sie müssen nur die grundlegenden Konfigurationsdaten angeben und können dadurch die SNA-Kommunikation schnell und einfach konfigurieren.

Die übrigen Verwaltungstools von Communications Server für Linux wie das Befehlszeilenverwaltungsprogramm und NOF-Anwendungsprogramme bieten Zugriff auf ein breiteres Spektrum von Konfigurationsparametern und -optionen als das Motif-Verwaltungsprogramm. In den meisten Fällen können Sie jedoch alle erforderlichen Konfigurationsschritte im Motif-Verwaltungsprogramm ausführen, denn es zeigt die für die Konfiguration benötigten Schlüsselfelder an und verdeckt die Felder, die von den meisten Benutzern nicht geändert werden müssen. Die vom Befehlszeilenverwaltungsprogramm vorgegebenen Standardwerte können sich von den vom Motif-Verwaltungsprogramm angegebenen unterscheiden, denn das Motif-Verwaltungsprogramm kann Werte ausgehend vom Kontext der von Ihnen ausgeführten Konfigurationsaufgabe intelligenter auswählen.

Falls Sie die zusätzlichen Funktionen benötigen, können Sie ungeachtet dessen die Basiskonfiguration mit dem Motif-Verwaltungsprogramm einrichten und die übrigen Verwaltungstools zur Angabe der zusätzlichen Funktionen verwenden. Wenn Sie später die geänderte Konfiguration mit dem Motif-Verwaltungsprogramm verwalten, merkt sich das Programm die Änderungen, die Sie mit anderen Tools vorgenommen haben, obwohl die von Ihnen konfigurierten Funktionen im Motif-Verwaltungsprogramm nicht angezeigt werden.

### Befehlszeilenverwaltungsprogramm

Mit Hilfe des Befehlszeilenverwaltungsprogramms **snaadmin** können Sie Befehle für die Verwaltung einzelner CS-Linux-Ressourcen absetzen. Sie können **snaadmin** direkt an einer Linux-Eingabeaufforderung ausführen oder in eine Shell-Prozedur integrieren.

Sie können Befehle auf dem CS-Linux-Knoten absetzen, um die Ressourcen des Knotens zu verwalten, in der SNA-Netzdatendatei, um Haupt- und Ausweichserver zu verwalten, oder in der Domänenkonfigurationsdatei, um die Domänenressourcen zu verwalten.

Auf einem Server können alle Verwaltungsbefehle abgesetzt werden. Für das Absetzen von Befehlen auf einem IBM Remote API Client gelten jedoch Einschränkungen.

- Auf Windows-Clients gibt es kein Programm **snaadmin**, so dass keine Befehle abgesetzt werden können.
- Auf AIX- und Linux-Clients können Sie jeden **query**- oder **status**-Befehl absetzen. Für einige andere Verwaltungsbefehle, die in der Veröffentlichung "Communications Server for Linux Administration Command Reference" definiert sind, ist explizit angegeben, dass Sie von einem IBM Remote API Client abgesetzt werden können. Andernfalls sind diese Befehle nur von einem Server aus verfügbar.

Hilfe zum Befehlszeilenverwaltungsprogramm können Sie mit den folgenden Befehlen aufrufen:

- **snaadmin -h** stellt Basisinformationen zum Befehlszeilenverwaltungsprogramm und Hilfetexte zur Verwendung dieses Programms bereit.

- **snaadmin -h -d** ruft eine Liste von Befehlen auf, die das Programm **snaadmin** ausführen kann.
- **snaadmin -h *Befehl*** ruft einen Hilfetext für den benannten *Befehl* auf.
- **snaadmin -h -d *Befehl*** stellt ausführliche Hilfeinformationen zum benannten *Befehl* und eine Liste der für diesen Befehl verfügbaren Konfigurationsparameter bereit.

Weitere Informationen hierzu finden Sie in der Veröffentlichung *Communications Server for Linux Administration Command Reference*.

### Funktion für ferne Befehle

Die Einrichtung für ferne Befehle (RCF) stellt die folgenden Funktionen zur Unterstützung der Verwaltung von Communications Server für Linux über eine NetView-Konsole auf einem Host bereit:

- Mit der Servicepunkt-Befehlsfunktion (SPCF, Service Point Command Facility) kann ein Bediener Communications Server für Linux über NetView verwalten, indem er auf einer NetView-Konsole CS-Linux-Verwaltungsbefehle absetzt.
- Mit der UNIX-Befehlsfunktion (UCF, UNIX Command Facility) kann der NetView-Bediener auf dem CS-Linux-Computer Linux-Standardbefehle absetzen.

Weitere Informationen zu RCF finden Sie in Kapitel 9, „Communications Server für Linux mit NetView verwalten“, auf Seite 127.

### Konfigurationsdateien

Die Konfigurationsdaten für das CS-Linux-System sind in den folgenden Textdateien enthalten:

#### Knotenkonfigurationsdatei

Die Datei **/etc/opt/ibm/sna/sna\_node.cfg** enthält Informationen zu den CS-Linux-Knotenressourcen für einen bestimmten Knoten. Diese Datei befindet sich auf dem Computer, auf dem der Knoten aktiv ist. In dieser Datei sind Informationen zu den Ressourcen des Knotens und zu Ressourcen, die beim Starten von Communications Server für Linux aktiv sind, gespeichert.

Die Datei stellt eine erste Definition der verfügbaren Ressourcen bereit. Bei veränderten Anforderungen können Sie die Ressourcen des aktiven Knotens mit den übrigen Verwaltungstools ändern. Alle von Ihnen vorgenommenen Änderungen werden automatisch in der Datei gespeichert, so dass Sie die geänderte Konfiguration nach dem Stoppen und erneuten Starten des Knotens wieder verwenden können.

#### Domänenkonfigurationsdatei

Die Datei **/etc/opt/ibm/sna/sna\_domn.cfg** enthält Informationen zu den CS-Linux-Domänenressourcen (Ressourcen, die keinem bestimmten lokalen Knoten zugeordnet sind). Die Master-Kopie dieser Datei befindet sich auf dem Hauptserver.

#### Datendatei für aufrufbare TPs

Die Datei **/etc/opt/ibm/sna/sna\_tps** enthält Informationen, die Communications Server für Linux für das Starten von aufrufbaren Transaktionsprogrammen (Ziel-TPs) benötigt, und unter Umständen weitere Informationen (wie die für den Zugriff auf das TP erforderliche Sicherheitsstufe). Diese Datei befindet sich auf dem Computer, auf dem die TPs ausgeführt werden.

Ausführliche Informationen zu dieser Datei finden Sie im Abschnitt „TPs definieren“ auf Seite 93.

## Verwaltung von Communications Server für Linux im Überblick

Sie können die Konfiguration mit dem Motif-Verwaltungsprogramm, mit dem Befehlszeilenverwaltungsprogramm oder mit der API NOF ändern. Alle genannten Tools nehmen ggf. die erforderlichen Änderungen an der Knotenkonfigurationsdatei oder der Domänenkonfigurationsdatei vor. Da die Konfigurationsdaten als unverschlüsselter Text gespeichert werden, können Sie die Datei direkt mit einem Standard-ASCII-Editor wie **vi** oder über eine Shell-Prozedur mit Linux-Dienstprogrammen wie **awk** oder **sed** ändern. Die Bearbeitung von Konfigurationsdateien mit einem Texteditor muss erfolgen, **bevor** Communications Server für Linux gestartet wird. Weitere Informationen zum Format der Konfigurationsdateien von Communications Server für Linux finden Sie in der Veröffentlichung *Communications Server for Linux Administration Command Reference*.

**Anmerkung:** Das Konfigurieren von Communications Server für Linux ist ein dynamischer Prozess. Es ist nicht notwendig, dass Sie vor dem Starten von Communications Server für Linux die gesamte Konfiguration definieren. Die Konfigurationsdatei stellt eine erste Definition der verfügbaren Ressourcen bereit. Wenn Communications Server für Linux aktiv ist, können Sie bei Bedarf Ressourcen hinzufügen, löschen oder ändern. Communications Server für Linux speichert die aktuelle Definition, so dass Sie sie nach einem notwendigen Neustart des Systems wieder verwenden können.

Die folgenden Dateien enthalten Informationen zum Client-Server-Netz mit Communications Server für Linux:

### SNA-Netzdatendatei

Die Datei **/etc/opt/ibm/sna/sna.net** enthält Informationen darüber, welcher Server der Hauptserver ist und welche Server als Ausweichserver eingesetzt werden können. Diese Binärdatei befindet sich auf dem Hauptserver. Sie können den Inhalt dieser Datei mit den Verwaltungsprogrammen oder der API NOF ändern.

Ausführliche Informationen zu dieser Datei und zum Modifizieren dieser Datei finden Sie im Abschnitt „Client-Server-Funktionen konfigurieren“ auf Seite 57.

### Clientnetzdatendatei

Die Datei **sna\_clnt.net** enthält Informationen darüber, wie auf die für einen IBM Remote API Client erforderlichen CS-Linux-Server zugegriffen werden kann. Diese Textdatei befindet sich auf dem Clientcomputer. Sie können den Inhalt dieser Datei mit einem Standard-ASCII-Texteditor ändern.

Ausführliche Informationen zu dieser Datei und zum Modifizieren dieser Datei finden Sie im Abschnitt „Netzdatendatei des Clients (sna\_clnt.net)“ auf Seite 160. Hinweise zum Konfigurieren der äquivalenten Informationen auf einem Windows-Client finden Sie in Kapitel 10, „Client-Server-Systeme mit Communications Server für Linux verwalten“, auf Seite 137.

### NOF-Anwendungen

Die API NOF von Communications Server für Linux stellt die gleichen Verwaltungsfunktionen wie das Befehlszeilenverwaltungsprogramm bereit und ermöglicht Ihnen, die CS-Linux-Ressourcen zu definieren und zu verwalten. Sie können auch eigene Programme für die Verwaltung von Communications Server für Linux schreiben.

Weitere Informationen hierzu finden Sie in der Veröffentlichung *Communications Server for Linux NOF Programmer's Guide*.

## Diagnosetools

Communications Server für Linux stellt verschiedene Diagnosetools bereit, die Sie beim Bestimmen und Beheben von Fehlern, die während des Betriebs von Communications Server für Linux auftreten, unterstützen.

- Jede Komponente, die einen Fehler oder eine Ausnahme (die möglicherweise auf die Ursache eines Fehlers hinweist) feststellt, schreibt einen Eintrag in eine Fehlerprotokolldatei. Zusätzlich können alle bedeutsamen Systemereignisse in einer Prüfprotokolldatei aufgezeichnet werden. Sie legen fest, welche Arten von Ereignissen (Fehler, Ausnahmen oder Prüfungen) protokolliert werden sollen. In einer Client-Server-Netzkonfiguration können Sie globale Einstellungen für die auf allen Servern zu erfassenden Arten von Ereignissen angeben und diese dann bei Bedarf für einzelne Server überschreiben.
- Communications Server für Linux verwaltet außerdem eine Auslastungsprotokolldatei, in der Informationen zur aktuellen und maximalen Nutzung der CS-Linux-Ressourcen erfasst werden.
- Sie können für die Dateien mit den Protokollinformationen die Namen und Verzeichnisse angeben. Bei Bedarf können Sie sogar Fehler- und Prüfprotokollinformationen an dieselbe Datei senden. Auf einem Client-Server-System können Sie Nachrichten von allen Servern an eine zentrale Protokolldatei auf einem Server senden (zentrale Protokollierung) oder Protokollnachrichten an separate Dateien auf den einzelnen Servern senden.
- Protokolldateien werden als Textdateien generiert und können mit einem Standard-ASCII-Texteditor, z. B. **vi**, angezeigt werden.
- Sie können die vollständige Protokollierung (bei der in der Protokolldatei für jede Nachricht Details zur Ursache der Protokollierung und alle erforderlichen Aktionen angegeben sind) oder die kurze Protokollierung (bei der nur der Nachrichtentext und eine kurze Zusammenfassung der Ursache der Protokollierung angegeben sind) auswählen. Wenn Sie mit der kurzen Protokollierung arbeiten und weitere Informationen benötigen, können Sie mit dem Befehlszeilendienstprogramm **snahelp** für eine bestimmte Nachrichtennummer die vollständige Anzeige der Ursache und aller notwendigen Maßnahmen aufrufen.
- Wenn Sie feststellen, dass ein bestimmtes Ereignis häufig eintritt und sich die Protokolldatei demzufolge mit vielen Instanzen derselben Protokollnachricht füllt, können Sie einen Filter setzen, um anzugeben, dass bestimmte Nachrichten nur einmal protokolliert werden sollen. Alle nachfolgenden Instanzen derselben Protokollnachricht werden dann ignoriert und nicht in die Protokolldatei geschrieben.
- Bei einigen Fehlerbedingungen schreibt Communications Server für Linux nicht nur eine Fehlermeldung in die Fehlerprotokolldatei, sondern sendet außerdem eine Nachricht an die Linux-Konsole, um den Bediener zu warnen.
- Viele Komponenten können eine Trace-Datei generieren, in der die Aktivitäten der Komponente aufgezeichnet werden. Der Trace setzt die Leistung der CS-Linux-Komponenten herab und ist deshalb normalerweise inaktiviert.
- Mit den Befehlszeilendienstprogrammen können Sie bestimmte Informationen aus den Trace-Dateien filtern und extrahieren und diese dann formatieren, um ihren Inhalt zu interpretieren oder eine Zusammenfassung des Nachrichtenflusses zu erstellen. Die formatierten Ausgabedateien können Sie mit einem Standard-ASCII-Texteditor, z. B. **vi**, anzeigen.
- Communications Server für Linux kann Alerts generieren und an das Programm NetView auf einem Host senden. Hierbei kann es sich um die folgenden Alerts handeln:
  - Verbindungs-Alerts von Konnektivitätskomponenten, die Informationen zu Verbindungsproblemen liefern

## Verwaltung von Communications Server für Linux im Überblick

- Alerts, die von Anwendungsprogrammen über die API MS bereitgestellt werden

Informationen zu den Protokollnachrichten von Communications Server für Linux, zur Verwendung der CS-Linux-Trace-Einrichtungen und zur Interpretation von Trace-Dateien finden Sie im *Communications Server for Linux Diagnostics Guide*.

Informationen zur Verwendung der API MS enthält der *Communications Server for Linux MS Programmer's Guide*.

### Berechtigungen für die Verwaltung

Die Verwaltungstools von Communications Server für Linux sind für eine eingeschränkte Gruppe von „SNA-Administratoren“ bestimmt, die berechtigt sind, SNA-Ressourcen zu verwalten. Deshalb ist der Systemadministrator der Gruppe sna mit der Anmelde-ID root Eigner der ausführbaren Dateien. Nur Benutzer der Gruppe sna sind berechtigt, CS-Linux-Ressourcen zu ändern, zu aktivieren oder zu inaktivieren. Jeder Benutzer, der die Berechtigungen eines SNA-Administrators haben soll, muss zu dieser Gruppe gehören.

Wenn Sie die Standardinstallation von Communications Server für Linux ausgeführt haben, können Benutzer, die nicht zur Gruppe sna gehören, keines der Verwaltungstools von Communications Server für Linux ausführen. Bei Bedarf können Sie diesen Benutzern die Ausführung der Tools im Lesezugriff gestatten, damit sie sich Konfigurationsdaten und Statusinformationen ansehen können. Diese Benutzer können dann jedoch keine Ressourcen ändern, aktivieren oder inaktivieren. Verwenden Sie **chmod**, um einem Benutzer für die gewünschte ausführbare Datei die Ausführungsberechtigung im Lesezugriff zu erteilen.

Verwaltungstool	Ausführbare Datei(en)
Motif-Verwaltungsprogramm	<code>/opt/ibm/sna/bin/X11/xsnaadmin</code>
Befehlszeilenverwaltungsprogramm	<code>/opt/ibm/sna/bin/snaadmin</code>

Jeder Benutzer kann dann das entsprechende Verwaltungstool ausführen und Informationen anzeigen, und trotzdem kann Communications Server für Linux gewährleisten, dass Benutzer, die nicht zur Gruppe sna gehören, Ressourcen nicht ändern, aktivieren oder inaktivieren können.

**Anmerkung:** Wenn Sie die Dateiberechtigungen wie oben beschrieben geändert haben, müssen Sie diese Prozedur nach Installation von vorläufigen Programmkorrekturen oder neuen Releases zu Communications Server für Linux nicht wiederholen.

---

## CS-Linux-Konfiguration planen

Es ist sehr wichtig, dass Sie jede Konfigurationsänderung vor ihrer Ausführung sorgfältig planen. Von Ihnen vorgenommene Änderungen können die Arbeit der Benutzer des lokalen Knotens und unter Umständen sogar aller Benutzer im gesamten Netz unterbrechen.

Eine gute Hilfe ist eine grafische Übersicht über alle Änderungen, die Sie an der Topologie des Netzes vornehmen. Wenn Sie Verbindungen zu anderen Knoten hinzufügen oder entfernen, zeichnen Sie eine Skizze des lokalen Knotens und der

anderen Knoten. Über das Motif-Verwaltungsprogramm können Sie Informationen zu allen vorhandenen Verbindungen aufrufen und diese in Ihre grafische Übersicht aufnehmen.

Wenn Sie Ihre Übersicht durch neue Ressourcen ergänzen, können Sie leicht feststellen, ob diese Ressourcen eventuell schon vorhanden sind oder Namenskonflikte auftreten. Ihre grafische Übersicht unterstützt Sie bei der Entscheidung, welche Ressourcen entfernt werden müssen, und hilft verhindern, dass Sie unbedingt erforderliche Ressourcen löschen.

Wenn Sie ein Client-Server-System mit CS-Linux mit mehreren Knoten konfigurieren, vergewissern Sie sich, dass alle CS-Linux-Knoten mit den zugehörigen Konnektivitätsressourcen in Ihre grafische Übersicht aufgenommen wurden.

Wenn Sie bestimmen, welche Änderungen Sie vornehmen wollen, können Sie gleich die erforderlichen Konfigurationsdaten zusammenstellen. Mit den Aufgabenblättern in der Onlinehilfe des Motif-Verwaltungsprogramms oder den im Abschnitt „Planungsblätter“ beschriebenen Planungsblättern können Sie feststellen, welche Konfigurationsdaten Sie für bestimmte CS-Linux-Funktionen erfassen müssen.

### Planungsblätter

Bevor Sie mit dem Konfigurieren von Ressourcen für Communications Server für Linux beginnen, müssen Sie alle Konfigurationsdaten für die neuen Ressourcen zusammenstellen. Verwenden Sie zum Erfassen der Daten für eine bestimmte unterstützende Funktion oder Anwendung die in Anhang A, „Planungsblätter für die Konfiguration“, auf Seite 165 enthaltenen Planungsblätter.

Möglicherweise müssen Sie Informationen von verschiedenen Quellen anfordern, z. B. von Netzadministratoren, Hostadministratoren, Anwendungsprogrammierern und Endbenutzern.

Wenn Sie versuchen, eine Verbindung zu einem anderen Knoten herzustellen, ist der Administrator dieses Knotens Ihr erster und wichtigster Ansprechpartner. Der Administrator eines Knotens kann Ihnen Namen, Adressen und Kenndaten aller Ressourcen auf diesem Knoten nennen. Oft müssen Sie sicherstellen, dass auf dem lokalen und dem fernen Knoten übereinstimmende Konfigurationsparameter eingegeben werden.

### Aufgabenblätter

Die Onlinehilfe des Motif-Verwaltungsprogramms enthält Aufgabenblätter mit einer Anleitung für bestimmte Konfigurationaufgaben. In den Aufgabenblättern finden Sie Zeiger auf alle Hilfeanzeigen zu den Dialogen, in denen Sie die Konfigurationsdaten eingeben müssen. Mit diesen Aufgabenblättern können Sie die Hilfetexte durchblättern und genau feststellen, welche Daten zu erfassen sind.

Die Aufgabenblätter enthalten außerdem Verweise auf ausführlichere Hilfetexte zu den einzelnen Fenstern und Dialoganzeigen, die Sie zum Eingeben der Konfigurationsdaten verwenden müssen. In diesen Hilfeanzeigen ist jedes einzelne Feld, das Sie auswählen oder in dem Sie einen Wert eingeben müssen, erläutert.

### Communications Server für Linux auf dem lokalen System aktivieren und inaktivieren

In den folgenden Abschnitten wird erläutert, wie Sie Communications Server für Linux auf dem Linux-Server aktivieren und inaktivieren können.

Bevor sie die CS-Linux-Tools (z. B. das Motif-Verwaltungsprogramm) verwenden können, müssen Sie Communications Server für Linux aktivieren. Normalerweise wird Communications Server für Linux nach Installation der Software automatisch aktiviert. Sollte dies nicht der Fall sein, müssen Sie die Software manuell aktivieren.

#### Pfad zu CS-Linux-Programmen angeben

Ausführbare Programme von Communications Server für Linux sind in einem spezifischen Verzeichnis von Communications Server für Linux gespeichert. Wenn Sie die Programme ausführen wollen, müssen Sie den Pfad zu diesem Verzeichnis angeben. Sie können den Pfad angeben, indem Sie vor der ersten Ausführung der Programme das Verzeichnis zur Umgebungsvariablen PATH hinzufügen oder indem Sie bei jedem Aufruf der Programme den Verzeichnisnamen mit angeben.

Das Motif-Verwaltungsprogramm ist im Verzeichnis `/opt/ibm/sna/bin/X11` gespeichert. Die übrigen Programme befinden sich im Verzeichnis `/opt/ibm/sna/bin`. Wenn Sie diese Verzeichnisse zur Definition der Umgebungsvariablen PATH in Ihrer Datei `.login` oder `.profile` hinzufügen, findet Communications Server für Linux die Programme automatisch. Alternativ dazu können Sie wie in den folgenden Beispielen den Verzeichnisnamen angeben, wenn Sie das Programm ausführen:

```
/opt/ibm/sna/bin/sna start
```

```
/opt/ibm/sna/bin/snaadmin query_node
```

```
/opt/ibm/sna/bin/X11/xsnaadmin
```

Bei den in diesem Handbuch angegebenen Beispielbefehlszeilen wird davon ausgegangen, dass Sie die Verzeichnisse zur Umgebungsvariablen PATH hinzugefügt haben und die Verzeichnisnamen nicht jedes Mal eingeben müssen.

#### CS-Linux-Server aktivieren

Dieser Abschnitt beschreibt das Aktivieren von Communications Server für Linux auf einem Computer, der als Server (d. h. mit installierten SNA-Knotenkomponenten) installiert wurde. Wenn Sie Communications Server für Linux auf einem Client aktivieren möchten, lesen Sie den Abschnitt „Remote API Clients unter AIX oder Linux aktivieren“ auf Seite 159.

Bevor Sie den lokalen Knoten (lokal oder über einen fernen CS-Linux-Knoten) konfigurieren oder verwalten können, müssen Sie Communications Server für Linux auf dem lokalen System aktivieren.

Geben Sie zum Aktivieren der Software Communications Server für Linux an der Linux-Eingabeaufforderung den folgenden Befehl ein:

```
sna start [ -s ] [ -m Kernel-Speicherbegrenzung ] [ -t ]
```

**Anmerkung:** Wenn Sie den Befehl `sna start` angeben, verwendet die Software Communications Server für Linux das Verzeichnis, von dem aus Sie

## Communications Server für Linux auf dem lokalen System aktivieren und inaktivieren

den Befehl abgesetzt haben, als aktuelles Arbeitsverzeichnis und verwaltet offene Dateideskriptoren in diesem Verzeichnis. Deshalb können Sie das Dateisystem mit diesem Verzeichnis nicht abhängen, solange die Software Communications Server für Linux aktiv ist. Um Probleme zu vermeiden, sollten Sie die Software Communications Server für Linux von einem Verzeichnis in einem Dateisystem starten, das nicht abgehängt werden muss. Sie könnten beispielsweise mit `cd /` das Stammverzeichnis aufrufen, bevor Sie den Befehl **sna start** absetzen.

Bei der Installation von Communications Server für Linux aktualisiert das Installationsdienstprogramm automatisch die Startdatei `/etc/rc.d/init.d/snastart` und fügt einen Eintrag für den Befehl **sna start** hinzu. Auf diese Weise wird sichergestellt, dass Communications Server für Linux beim Systemstart automatisch gestartet wird. Falls Sie nicht möchten, dass Communications Server für Linux automatisch gestartet wird, können Sie diese Zeile löschen oder auf Kommentar setzen und dann die in diesem Abschnitt enthaltenen Anweisungen für das manuelle Aktivieren der Software Communications Server für Linux ausführen.

Der Befehl **sna start** kann mit folgenden Parametern und Optionen verwendet werden:

- s Gibt an, dass Communications Server für Linux keine Nachrichten auf die Systemkonsole schreiben soll. Wenn Sie diese Option nicht angeben, schreibt Communications Server für Linux vor Inaktivierung der Software Nachrichten auf die Konsole und gibt bestimmte Fehlerprotokollnachrichten sowohl in der Protokolldatei als auch auf der Konsole aus.
- m *Kernel-Speicherbegrenzung*  
Gibt die maximale Größe des von Communications Server für Linux verwendeten Kernel-Speichers in Kilobytes an. (Der Kernel-Speicher wird für interne Datenstrukturen benutzt.) Wenn eine CS-Linux-Komponente versucht, Kernel-Speicher zuzuordnen, und der allen Komponenten von Communications Server für Linux derzeit zugeordnete Speicher dadurch diesen Grenzwert überschreiten würde, schlägt der Zuordnungsversuch fehl.  
Wenn Sie diese Option nicht angeben, wird die Verwendung des Kernel-Speichers nicht begrenzt.
- t Aktiviert den IPS-Trace für alle Schnittstellen zwischen Kernel-Komponenten und den Client-Server-Trace. (Diese Option aktiviert nicht den DLC-Trace.) Die Durchführung eines Trace ermöglicht die Bestimmung von Fehlern, die bei der Initialisierung aufgetreten sind. Wenn Sie diese Option nicht verwenden, ist der Trace für alle Schnittstellen inaktiv. Sie können ihn bei Bedarf im Befehlszeilenverwaltungsprogramm **snaadmin** für bestimmte Schnittstellen aktivieren.  
Ein Trace für alle Schnittstellen setzt die Leistung der CS-Linux-Komponenten herab. Nach Aktivierung der Software können Sie den Trace im Befehlszeilenverwaltungsprogramm **snaadmin** für alle Schnittstellen stoppen, für die er nicht erforderlich ist. Weitere Informationen zum Durchführen von Traces finden Sie im *Communications Server for Linux Diagnostics Guide*.

Communications Server für Linux schreibt Nachrichten auf die Standardfehlerausgabe (in der Regel der Bildschirm Ihres Terminals) und gibt so an, ob die Initialisierung erfolgreich verläuft.

## Communications Server für Linux auf dem lokalen System aktivieren und inaktivieren

Wenn die Initialisierung fehlschlägt, enthält die Nachricht Informationen zur Fehlerursache und ggf. zusätzliche Informationen wie die Fehlermeldung des Betriebssystems Linux. Der in die Standardfehlerausgabe geschriebene Text enthält unter Umständen auch eine Nachricht mit dem Hinweis, dass Sie in der Fehlerprotokolldatei weitere Informationen finden können. Der Befehl **sna start** endet mit einem Exit-Code ungleich null, der die Art des Fehlers angibt.

Weitere Informationen zu den Werten von Exit-Codes finden Sie im *Communications Server for Linux Diagnostics Guide*.

### Erweiterte Optionen für den Befehl **sna start**

In einigen Fällen, insbesondere beim Testen neuer CS-Linux-Konfigurationen, kann es ratsam sein, Communications Server für Linux mit einer Konfiguration zu starten, die in einer temporären Datei (und nicht in den Standardkonfigurationsdateien **/etc/opt/ibm/sna/sna\_node.cfg** und **/etc/opt/ibm/sna/sna\_domn.cfg**) gespeichert wurde. Zu diesem Zweck können Sie den Befehl **sna start** mit den folgenden zusätzlichen Optionen verwenden:

- n** Knotenkonfigurationsdatei
- d** Domänenkonfigurationsdatei

*Knotenkonfigurationsdatei* steht hier für den vollständigen Pfadnamen der Datei, in der Sie die Knotenkonfiguration gespeichert haben (wenn sie nicht in **/etc/opt/ibm/sna/sna\_node.cfg** gespeichert ist). *Domänenkonfigurationsdatei* steht hier für den vollständigen Pfadnamen der Datei, in der Sie die Domänenkonfiguration gespeichert haben (wenn Sie nicht in **/etc/opt/ibm/sna/sna\_domn.cfg** gespeichert ist).

**Anmerkung:** Diese Optionen sind nicht zur allgemeinen Verwendung gedacht. Geben Sie sie nur an, wenn dies durch eine bestimmte Anforderung gerechtfertigt ist.

Wenn Communications Server für Linux mit diesen Optionen ausgeführt wird, funktioniert der Befehl **snagetpd** nicht ordnungsgemäß, da er immer Informationen aus den Standardkonfigurationsdateien verwendet. Stellen Sie vor der Verwendung von **snagetpd** sicher, dass Communications Server für Linux nicht mit diesen Optionen gestartet wurde, sondern mit den Standardkonfigurationsdateien ausgeführt wird.

## CS-Linux-Server inaktivieren

Durch Inaktivierung der Software Communications Server für Linux auf einem Server werden automatisch der CS-Linux-Knoten und die zugehörigen Konnektivitätskomponenten gestoppt. Nach Inaktivierung von Communications Server für Linux können alle Prozesse (z. B. 3270-Emulationsprogramme) auf diesem Server keine CS-Linux-Ressourcen mehr verwenden.

Sie sollten in der Regel nur einzelne Dienste stoppen, wenn diese von den Benutzern nicht mehr benötigt werden. Eine Inaktivierung des Systems ist nur anzuraten, wenn keine CS-Linux-Aktivität vorliegt. Bei Inaktivierung der Software Communications Server für Linux auf einem Client wird der Zugriff aller auf dem Client ausgeführten Programme auf die Funktionen von Communications Server für Linux gestoppt.

Wenn Sie Communications Server für Linux inaktivieren müssen, warnen Sie die Benutzer, dass Sie Communications Server für Linux stoppen wollen, und geben Sie ihnen Zeit, die gerade laufenden Aktivitäten vor Inaktivierung der Software zu

## Communications Server für Linux auf dem lokalen System aktivieren und inaktivieren

beenden. Einzelangaben zu aktiven Benutzern können Sie mit dem Motif-Verwaltungsprogramm oder dem Befehlszeilenverwaltungsprogramm anzeigen.

Benutzt ein 3270-Emulationsprogramm LUs auf dem Knoten, wenn Sie die Software Communications Server für Linux inaktivieren, werden alle 3270-Emulations-sitzungen, die diese LUs benutzen, beendet. Das Programm wird weiter ausgeführt, doch der Benutzer kann die Sitzungen erst wieder verwenden, wenn die Software reaktiviert wurde. Anwendungen, die die API APPC, CSV, LUA, NOF oder MS benutzen, werden durch einen Rückkehrcode COMM\_SUBSYSTEM\_ABENDED und CPI-C-Anwendungen durch einen Rückkehrcode CM\_PRODUCT\_SPECIFIC\_ERROR benachrichtigt.

Geben Sie zum Inaktivieren der Software Communications Server für Linux an der Linux-Eingabeaufforderung den folgenden Befehl ein:

```
sna stop
```

Wenn Communications Server für Linux erfolgreich inaktiviert wurde, gibt **sna stop** den Exit-Code 0 zurück. Jeder andere Exit-Code gibt an, dass ein Fehler aufgetreten ist und die Software Communications Server für Linux nicht inaktiviert wurde. Weitere Informationen zu den Werten von Exit-Codes finden Sie im *Communications Server for Linux Diagnostics Guide*.

---

## Motif-Verwaltungsprogramm verwenden

Das Motif-Verwaltungsprogramm stellt eine benutzerfreundliche Schnittstelle für das Konfigurieren von Communications Server für Linux bereit. Dieses Programm wird für die Verwaltung von Communications Server für Linux empfohlen, denn es führt Sie durch den Konfigurationsprozess und reduziert die Informationen, die Sie für das Erstellen einer funktionsfähigen Konfiguration eingeben müssen, auf ein Minimum.

Außerdem können Sie mit dem Motif-Verwaltungsprogramm ein aktives CS-Linux-System verwalten. Das Verwaltungsprogramm bietet Ihnen die Möglichkeit, die Konfiguration zu ändern, während Communications Server für Linux aktiv ist. Sie können Ressourcen hinzufügen, ändern und entfernen (in den meisten Fällen sogar, wenn der Knoten und seine Ressourcen aktiv sind) und den Betrieb sofort mit der geänderten Konfiguration fortsetzen.

Das Motif-Verwaltungsprogramm zeigt aktuelle Statusinformationen über die Schnittstelle an, die auch für die Konfiguration verwendet wird, und erleichtert so den Zugriff auf Statusinformationen zu den Domänen- und Knotenressourcen.

Alternativ dazu können Sie für die Konfiguration und Systemverwaltung CS-Linux-Befehle verwenden. Eine Zusammenfassung der Konfigurations- und Verwaltungsbefehle finden Sie im Abschnitt „Befehlszeilenverwaltungsprogramm verwenden“ auf Seite 55.

## Motif-Verwaltungsprogramm aufrufen

Stellen Sie vor Verwendung des Motif-Verwaltungsprogramms für Communications Server für Linux sicher, dass Communications Server für Linux wie im Abschnitt „CS-Linux-Server aktivieren“ auf Seite 40 beschrieben aktiviert wurde. (Wie bei allen X/Motif-Anwendungen müssen Sie gegebenenfalls die Umgebungsvariable DISPLAY definieren, um einen geeigneten X-Server anzugeben.)

## Motif-Verwaltungsprogramm verwenden

Wenn Sie das Motif-Verwaltungsprogramm im Hintergrund starten wollen, setzen Sie den folgenden Befehl ab:

```
xsnaadmin &
```

In einer Client-Server-Umgebung zeigt Communications Server für Linux das Domänenfenster an.

Für ein eigenständiges System zeigt Communications Server für Linux normalerweise das Knotenfenster an. Falls Sie den lokalen Knoten jedoch noch nicht konfiguriert haben, erscheint eine Hilfetextanzeige mit Hinweisen zur Erstkonfiguration des lokalen Knotens.

**Anmerkung:** Im vorliegenden Handbuch wird der Begriff Fenster für die Motif-Fenster verwendet, in denen Informationen zu CS-Linux-Ressourcen angezeigt werden. Ein Fenster kann ein oder mehrere Teilfenster enthalten. Eine Dialoganzeige ist ein Motif-Fenster, in dem Sie Informationen eingeben können.

## Ressourcenfenster

Im Domänenfenster und im Knotenfenster werden die meisten benötigten Informationen angezeigt. Diese Fenster ermöglichen außerdem einen schnellen Zugriff auf zusätzliche Informationen. Über diese Fenster können Sie schnell Angaben zu Ressourcen in Ihrem lokalen Netz aufrufen.

Das Domänenfenster zeigt alle definierten Knoten an und stellt Optionen für das Hinzufügen, Aktivieren und Inaktivieren von Knoten bereit. Durch doppeltes Klicken auf einen Knoten wird das Knotenfenster für diesen Knoten aufgerufen.

Im Knotenfenster werden alle wichtigen Ressourcen für einen bestimmten Knoten angezeigt.

Die Menüs des Domänenfensters und des Knotenfensters stellen folgende Funktionen bereit:

### Auswahl

Die Funktionen dieses Menüs beziehen sich auf den Knoten, der derzeit im Domänenfenster ausgewählt ist, oder auf das zurzeit im Knotenfenster ausgewählte Element. Über dieses Menü können Sie den Knoten aktivieren oder inaktivieren oder das zugehörige Knotenfenster aufrufen. Wenn Sie im Knotenfenster ein Element auswählen, können Sie dieses Element über die Einträge dieses Menüs steuern, ändern oder löschen oder im ausgewählten Teilfenster ein neues Element hinzufügen.

### Dienste

Dieses Menü gestattet den schnellen Zugriff auf alle Dialoganzeigen, die für das Konfigurieren des Knotens für allgemeine Aufgaben erforderlich sind. Über dieses Menü können Sie Ressourcen hinzufügen oder ändern oder Hilfetexte zu Konfigurations- und Verwaltungsaufgaben aufrufen.

### Diagnoseprogramm

Über die Einträge dieses Menüs können Sie die Protokollierung und die Trace-Durchführung steuern.

### Fenster

Mit Hilfe dieses Menüs können Sie schnell auf andere Fenster zugreifen. Dazu gehören unter anderem:

- Fenster für LU-Pools
- Fenster für CPI-C-Bestimmungsorte

In Abhängigkeit davon, welche Ressourcen und Optionen Sie ausgewählt haben, kann das Verwaltungsprogramm zusätzliche Ressourcenfenster, Konfigurationsdialoganzeigen oder Statusprotokolle anzeigen. Verfügbar sind unter anderem Kontextdialoganzeigen, in denen Sie eine bestimmte zu konfigurierende Ressource auswählen können, Bestätigungsdialoganzeigen und Nachrichtenanzeigen mit Rückmeldungen oder Fehlernachrichten. Jedes Fenster und jede Dialoganzeige stellt eine Hilfeoption bereit.

### Domänenfenster

Das Domänenfenster zeigt alle aktiven SNA-Knoten in der CS-Linux-Domäne für das von Ihnen verwendete System an. (Ein Knoten erscheint nur dann im Domänenfenster, wenn er Communications Server für Linux ausführt.) Jeder Knoten ist mit dem Namen des Systems angegeben. Das Domänenfenster zeigt außerdem den aktuellen Status aller Knoten in der Domäne an.

**Anmerkung:** Sollte ein Server nicht in der Liste der Knoten im Domänenfenster enthalten sein, überprüfen Sie, ob der Server eingeschaltet ist und die Software Communications Server für Linux auf dem Server ausgeführt wird. Ggf. können Sie die Software Communications Server für Linux auf diesem Knoten mit dem Befehl **sna start** starten (siehe Abschnitt „CS-Linux-Server aktivieren“ auf Seite 40).

Ein Knoten in einer Domäne ist immer als Konfigurationsserver für die Domäne angegeben. Neben diesem Knoten wird im Domänenfenster das Wort „Hauptserver“ angezeigt. Der Hauptkonfigurationsserver enthält immer Konfigurationsdaten für Domänenressourcen. Die Ausweichkonfigurationsserver sind mit dem Wort „Ausweichserver“ gekennzeichnet. Ausweichkonfigurationsserver enthalten Kopien der Konfigurationsdaten für Domänenressourcen.

Abb. 13 zeigt ein Beispiel für ein Domänenfenster.

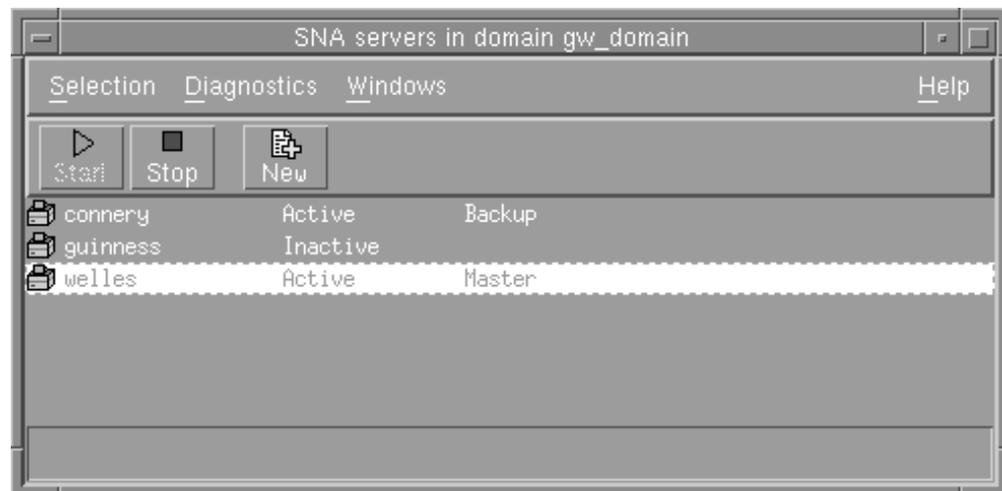


Abbildung 13. Domänenfenster für Communications Server für Linux

Wenn aktive Knoten in der Domäne (auf denen Communications Server für Linux ausgeführt wird) nicht konfiguriert sind, fordert Communications Server für Linux Sie zum Konfigurieren der Knoten auf.

## Motif-Verwaltungsprogramm verwenden

**Anmerkung:** Das Domänenfenster listet keine IBM Remote API Clients auf. Clients verwenden die Ressourcen von CS-Linux-Servern (SNA-Knoten), um auf SNA-Ressourcen zuzugreifen.

Im Domänenfenster können Sie die folgenden Verwaltungsaufgaben ausführen:

### **Knoten in der Domäne aktivieren oder inaktivieren**

Wählen Sie in diesem Fenster die Zeile für den Knoten aus, und klicken Sie auf die Schaltfläche **Starten** oder **Stoppen**. (Alternativ dazu können Sie auf die Zeile für den Knoten klicken und dann im Menü **Auswahl** den Eintrag **Knoten starten** oder **Knoten stoppen** auswählen.)

### **Einen bestimmten Knoten verwalten**

Klicken Sie im Domänenfenster doppelt auf die Zeile für den Knoten. (Alternativ dazu können Sie auf die Zeile für den Knoten klicken und dann im Menü **Auswahl** den Eintrag **Merkmale** auswählen. Sie können auch im Menü **Fenster** das Fenster für den Knoten auswählen.)

Wenn Sie den zu verwaltenden Knoten ausgewählt haben, zeigt Communications Server für Linux das in Abb. 14 auf Seite 47 dargestellte Knotenfenster an. (Für ein eigenständiges System zeigt Communications Server für Linux kein Domänenfenster an, weil die Domäne nur einen Knoten enthält. Stattdessen zeigt Communications Server für Linux nach dem Start des Verwaltungsprogramms direkt das Knotenfenster an.)

### **Knoten zur Serverliste für die Domäne hinzufügen**

Klicken Sie auf die Zeile für den Knoten, und wählen Sie im Menü **Auswahl** den Eintrag **Als Konfigurationsserver festlegen** aus.

### **Knoten aus der Serverliste für die Domäne entfernen**

Klicken Sie auf die Zeile für den Knoten, und wählen Sie im Menü **Auswahl** den Eintrag **Konfigurationsserver entfernen** aus.

### **Protokollierung für alle Knoten in der Domäne konfigurieren**

Wählen Sie im Menü **Diagnoseprogramm** den Eintrag **Protokollierung** aus.

### **Trace für einen bestimmten Knoten aktivieren oder inaktivieren**

Klicken Sie auf die Zeile für den Knoten, und wählen Sie im Menü **Diagnoseprogramm** den Eintrag **Trace für ausgewählten Knoten** aus.

### **Informationen zu Domänenressourcen abrufen**

Wählen Sie im Menü **Fenster** eine der Optionen aus. Neben den gemeinsam genutzten Domänenressourcen listet das Menü **Fenster** auch die einzelnen Knotenfenster für die Domäne auf.

## **Knotenfenster**

Abb. 14 auf Seite 47 zeigt ein Beispiel für ein Knotenfenster. In der Titelleiste ist der Name des Systems angegeben.

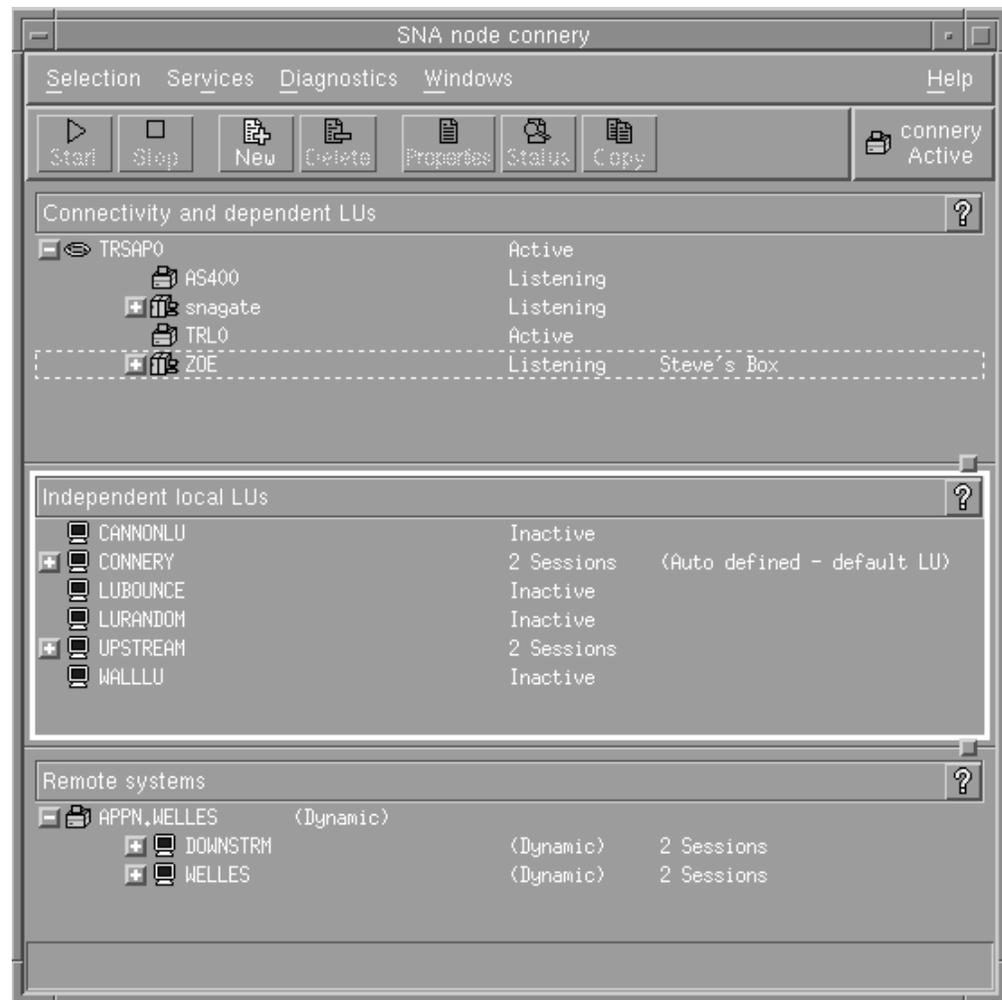


Abbildung 14. Knotenfenster

Im Knotenfenster können Sie alle Ressourcen und Komponenten für den CS-Linux-Knoten hinzufügen, löschen, ändern und verwalten. Die Darstellung der Ressourcen im Fenster gibt Aufschluss über die Anordnungsbeziehungen zwischen Ressourcen und ermöglicht Ihnen zu steuern, welche Ressourcen angezeigt werden.

Das Knotenfeld oben rechts im Knotenfenster gibt an, ob der Knoten Aktiv oder Inaktiv ist.

Alle für den Knoten definierten Ports, lokalen LUs und fernen LUs werden immer angezeigt. Im Knotenfenster wird jede Verbindungsstation unter dem zugehörigen Eltern-Port und jede abhängige LU unter der zugehörigen Elternverbindungsstation angezeigt. Darüber hinaus werden Partner-LUs unter lokalen LUs und fernen Knoten angezeigt.

Der Hauptteil des Knotenfensters ist für die verschiedenen Arten von Knotenressourcen in die folgenden Teilfenster unterteilt:

### Konnektivitätsteilfenster

Im oberen Teilfenster des Knotenfensters sind die Konnektivitätsressourcen aufgelistet. Dazu gehören Ports, Verbindungsstationen oder PUs der einzel-

## Motif-Verwaltungsprogramm verwenden

nen Ports und abhängige LUs einer bestimmten Verbindungsstation oder PU. Dieses Fenster zeigt aktuelle Statusinformationen zu jeder einzelnen Ressource an.

### Teilfenster für unabhängige lokale LUs

Im mittleren Teilfenster werden die unabhängigen LUs des Knotens angezeigt. In diesem Fenster werden zu jeder LU Informationen zu den die LU verwendenden Sitzungen angezeigt.

### Teilfenster für ferne Systeme

Das untere Teilfenster enthält Informationen zu fernen Knoten und Partner-LUs. Darüber hinaus zeigt es Sitzungsinformationen zu jedem fernen Knoten und zu jeder Partner-LU an.

Das Größenverhältnis der Teilfenster untereinander können Sie ändern, indem Sie auf die Grenzen zwischen den Teilfenstern klicken und diese ziehen.

Zum Auswählen eines Teilfensters müssen Sie darauf klicken. Bestimmte Ressourcen innerhalb eines Teilfensters können Sie auswählen, indem Sie in die Zeile für die Ressource klicken. Wenn Sie die Konfiguration einer Komponente anzeigen oder ändern wollen, müssen Sie doppelt auf die Komponente klicken. (Über die Schaltflächen und Menüs dieses Fensters können Sie außerdem auf die Konfigurationsdaten bestimmter Ressourcen zugreifen.)

Zu den Informationen, die zu jeder aufgelisteten Komponente angezeigt werden, gehören auch Informationen zu den zugehörigen Ressourcen. Verbindungsstationen sind beispielsweise unter dem Port, zu dem sie gehören, aufgelistet. Werden die Ressourcen zu einem Eintrag nicht angezeigt, können Sie neben dem Eintrag auf die Schaltfläche **Einblenden**



klicken. Umgekehrt können Sie auf die Schaltfläche **Ausblenden**



klicken, um die Anzeige der Ressourcen für einen Eintrag auszublenden.

Im Knotenfenster können Sie die folgenden Verwaltungsaufgaben ausführen:

### Ressource aktivieren oder inaktivieren

Wählen Sie die Ressource aus, und klicken Sie auf die Schaltfläche **Starten** oder **Stoppen**. (Alternativ dazu können Sie im Menü **Auswahl** die Option **Eintrag starten** oder **Eintrag stoppen** auswählen.)

### Neue Ressource für eine Komponente hinzufügen

Wählen Sie die Komponente aus, und klicken Sie auf die Schaltfläche **Neu** (oder wählen Sie im Menü **Auswahl** die Option **Neu** aus). Wenn Sie beispielsweise eine Verbindungsstation für einen Port hinzufügen wollen, wählen Sie den Port aus, und klicken Sie auf die Schaltfläche **Neu**.

### Ressource löschen

Wählen Sie die Ressource aus, und klicken Sie auf die Schaltfläche **Löschen** (oder wählen Sie im Menü **Auswahl** die Option **Löschen** aus).

### Konfiguration für eine Ressource anzeigen oder ändern

Wählen Sie die Ressource aus, und klicken Sie auf die Schaltfläche **Merkmale** (oder wählen Sie im Menü **Auswahl** die Option **Merkmale** aus).

### Statusinformationen für eine Ressource abrufen

Wählen Sie die Ressource aus, und klicken Sie auf die Schaltfläche **Status** (oder wählen Sie im Menü **Auswahl** die Option **Status** aus).

### Konfiguration für eine Ressource kopieren

Wählen Sie die Ressource aus, und klicken Sie auf die Schaltfläche **Kopieren** (oder wählen Sie im Menü **Auswahl** die Option **Kopieren** aus).

Darüber hinaus können Sie im Menü **Dienste** bestimmte Konfigurationsaufgaben für den Knoten auswählen, über das Menü **Diagnoseprogramm** die Protokollierung (für die Domäne) und den Trace (für den Knoten) steuern und durch Auswahl eines Eintrags im Menü **Fenster** Domänenressourcen anzeigen oder ändern.

### Ressourceneinträge

Die Darstellung der Ressourcen in einem Fenster gibt Aufschluss über die Anordnungsbeziehungen zwischen den Ressourcen.

Wenn einem Element Kindelemente zugeordnet sind, wird neben diesem Element die Schaltfläche **Einblenden** oder **Ausblenden** angezeigt. Die Schaltfläche **Einblenden** zeigt an, dass die zugehörigen Kindelemente verdeckt sind. Wenn Sie auf die Schaltfläche **Einblenden** klicken, werden diese angezeigt. Die Schaltfläche **Ausblenden** zeigt an, dass die Kindelemente angezeigt werden. Wenn Sie auf die Schaltfläche **Ausblenden** klicken, werden diese verdeckt. Wird neben einem Eintrag keine der Schaltflächen angezeigt, sind diesem Eintrag keine Kindressourcen zugeordnet.

Nehmen wir z. B. an, einem bestimmten Port ist eine Verbindungsstation zugeordnet. Im Konnektivitätsteilfenster des Knotenfensters wird die Verbindungsstation zusammen mit allen anderen diesem Port zugeordneten Verbindungsstationen unter dem Eltern-Port angezeigt. Der Port wird immer angezeigt. Sie können jedoch auswählen, ob die Liste der zugeordneten Verbindungsstationen ebenfalls angezeigt oder verdeckt wird. In ähnlicher Weise kann die Liste von LUs, die einer Verbindungsstation zugeordnet sind, angezeigt oder verdeckt werden.

Eine Elternressource muss stets vor den Kindressourcen konfiguriert werden. Beim Löschen der Elternressource werden gleichzeitig alle zugehörigen Kindressourcen gelöscht.

### Schaltflächen der Funktionsleiste

Ressourcenfenster enthalten eine Funktionsleiste mit Schaltflächen, die das Ausführen allgemeiner Funktionen vereinfachen. Abb. 15 zeigt eine Funktionsleiste für Communications Server für Linux.



Abbildung 15. Funktionsleiste für Communications Server für Linux

In den Funktionsleisten der einzelnen Ressourcenfenster werden nicht alle Schaltflächen angezeigt. Wenn die über eine Schaltfläche aktivierte Operation für das derzeit ausgewählte Element ungültig ist (oder die Operation die Auswahl eines Elements erfordert und kein Element ausgewählt ist), wird der Umriss der Schaltfläche grau angezeigt, und die Funktion kann nicht ausgewählt werden. (Das Klicken auf die Schaltfläche bleibt ohne Wirkung.) In Ressourcenfenstern werden die folgenden Schaltflächen angezeigt:

## Motif-Verwaltungsprogramm verwenden



Aktiviert das ausgewählte Element



Inaktiviert das ausgewählte Element



Fügt einen neuen Ressourceneintrag hinzu (Im Knotenfenster können Sie eine Ressource im ausgewählten Teilfenster hinzufügen.)



Löscht die ausgewählten Ressourcen



Öffnet für das ausgewählte Element den Dialog, über den die Konfiguration des Elements angezeigt oder geändert werden kann



Kopiert das ausgewählte Element. Durch Klicken auf diese Schaltfläche wird eine Dialoganzeige aufgerufen, deren Felder eine Kopie der Konfiguration der ausgewählten Komponente enthalten. Zum Hinzufügen der neuen Ressource müssen Sie die Angaben vervollständigen (den Namen der neuen Komponente eingeben).



Zeigt den aktuellen Status des ausgewählten Elements an

Viele Ressourcen, z. B. Ports und Verbindungsstationen, können nicht geändert werden, solange sie aktiv sind. Sie können jedoch die Parameter einer aktiven Ressource anzeigen, indem Sie die Ressource auswählen und auf die Schaltfläche **Merkmale** klicken, um die zugehörige Dialoganzeige aufzurufen, oder auf die Schaltfläche **Status** klicken, um ausführliche Statusinformationen zu der Ressource anzuzeigen.

## Dialoganzeigen für Ressourcen

In den Dialoganzeigen für Ressourcen werden die aktuellen Konfigurationsdaten der Ressource angezeigt. Abb. 16 zeigt eine Beispieldialoganzeige für LUs der Typen 0-3.

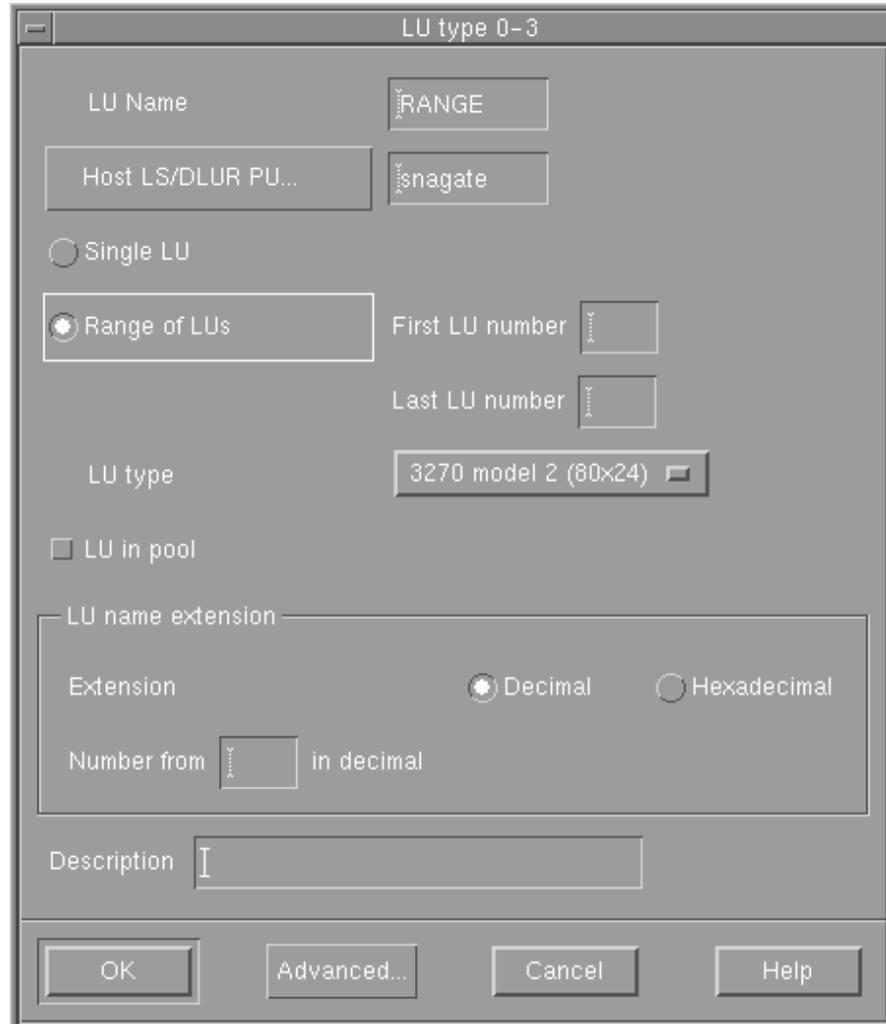


Abbildung 16. Beispieldialoganzeige

Die Dialoganzeigen für Ressourcen führen Sie durch den Konfigurationsprozess und geben Standardwerte vor, soweit dies möglich ist. Wenn Sie beispielsweise eine abhängige LU hinzufügen, trägt das Motif-Verwaltungsprogramm im Feld *LU-Nummer* automatisch eine für die von Ihnen angegebene Verbindungsstation verfügbare LU-Nummer ein. Geben Sie einen erforderlichen Wert nicht an, öffnet das Programm ein Nachrichtenfenster, in dem die von Ihnen bereitzustellenden Informationen angegeben sind.

## Motif-Verwaltungsprogramm verwenden

Die meisten Dialoganzeigen enthalten ein Feld *Beschreibung*. Die hier von Ihnen eingegebenen Informationen erscheinen in dem Fenster, in dem die Ressource angezeigt wird.

Falls Sie berechtigt sind, die Informationen in einer Ressourcendialoganzeige zu ändern, stehen in der Dialoganzeige die Schaltflächen **OK** und **Abbrechen** bereit, wenn Sie ein neues Element hinzufügen oder ein vorhandenes Element ändern. Klicken Sie, wenn Sie fertig sind, auf die Schaltfläche **OK**, oder klicken Sie auf die Schaltfläche **Abbrechen**, um die Anzeige ohne Änderung der Ressourcenkonfiguration zu verlassen.

Sollten Sie die Informationen in einer Ressourcendialoganzeige nicht ändern können (z. B., weil eine Ressource gerade aktiv ist und ihre Konfiguration deshalb nicht geändert werden kann), enthält die Dialoganzeige an Stelle der Schaltfläche **OK** die Schaltfläche **Schließen**. Klicken Sie auf diese Schaltfläche, wenn Sie die Informationen der Dialoganzeige gelesen haben.

Eine kontextbezogene Hilfe zur Dialoganzeige können Sie durch Klicken auf die Schaltfläche **Hilfe** aufrufen.

**Anmerkung:** In den wesentlichen Motif-Dialoganzeigen werden nur die Schlüsselkonfigurationsfelder angezeigt. Für erweiterte Felder gibt Communications Server für Linux Standardwerte vor. Wenn Sie auf die erweiterten Konfigurationsparameter zugreifen wollen, klicken Sie auf die Schaltfläche **Erweitert**. Sollten Sie erweiterte Parameter einstellen wollen, gehen Sie vor dem Aufrufen der erweiterten Dialoganzeige zunächst die Basisdialoganzeige durch, denn die erweiterte Anzeige kann sich in Abhängigkeit von den für Basisparameter eingegebenen Werten ändern. Hinweise zu den Feldern für die erweiterte Konfiguration finden Sie in den Onlinehilfetexten des Motif-Verwaltungsprogramms.

## Statusdialoganzeigen

Wenn Sie eine Ressource auswählen und auf die Schaltfläche **Status** klicken, ruft das Motif-Verwaltungsprogramm ausführliche Statusinformationen zu der Ressource auf. Sehen Sie sich hierzu Abb. 17 an.

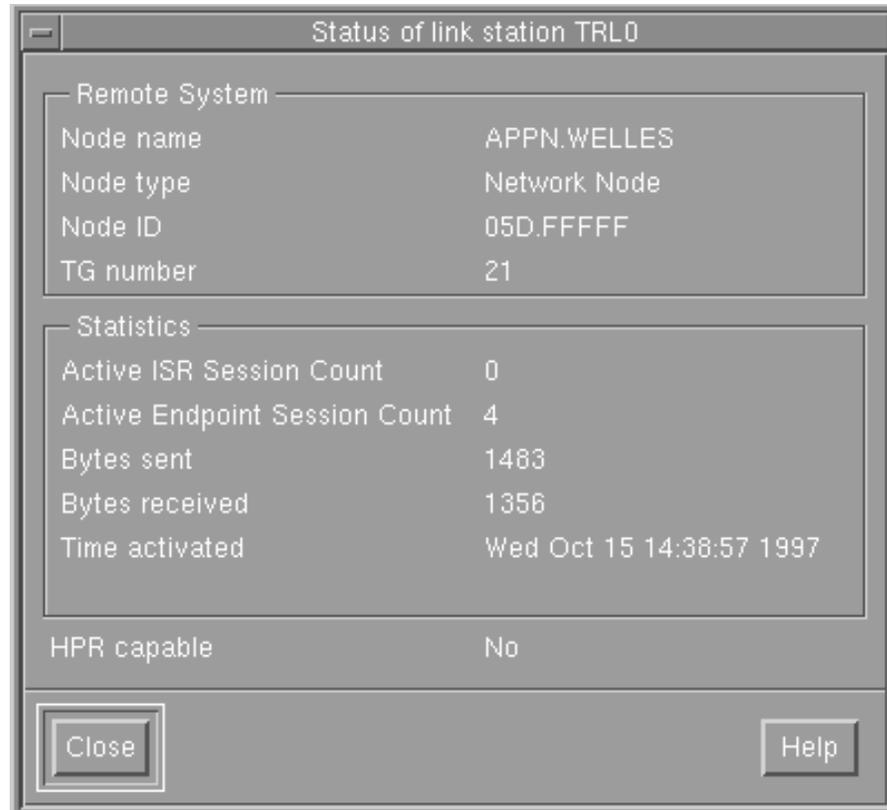


Abbildung 17. Beispiel für eine Statusdialoganzeige

Statusdialoganzeigen enthalten Informationen zum aktuellen Status der Ressource. Die Informationen werden auch während der Anzeige dynamisch aktualisiert.

### Hilfenfenster

Die Onlinehilfefunktion des Motif-Verwaltungsprogramms stellt eine ausführliche Anleitung für die einzelnen von Ihnen auszuführenden Konfigurationsaufgaben bereit. Insbesondere in den Aufgabenblättern ist jeder für das Konfigurieren einer bestimmten Ressource erforderliche Schritt genau erläutert. Das Aufgabenblatt für das Konfigurieren von Knotenparametern (dies ist immer der erste Schritt bei der CS-Linux-Konfiguration) kann wie in Abb. 18 gezeigt aufgerufen werden.

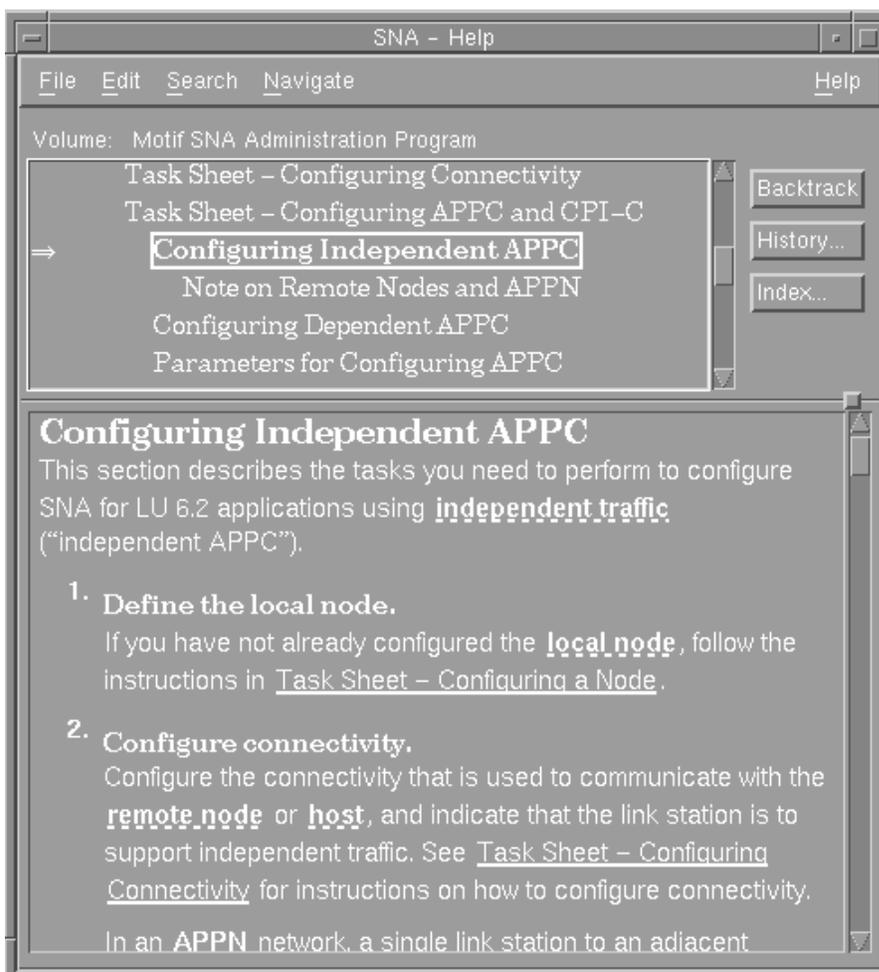


Abbildung 18. Beispiel für ein Hilfenfenster

Weitere Hilfenfenster sind für jedes Fenster und jede Dialoganzeige, für Fehlermeldungen und für SNA-Konzepte verfügbar.

## Befehlszeilenverwaltungsprogramm verwenden

Mit dem Befehlszeilenverwaltungsprogramm können Sie alle Konfigurationsparameter von Communications Server für Linux ändern. Sie können dieses Programm zum Konfigurieren aller im Motif-Verwaltungsprogramm angezeigten Ressourcen verwenden und Konfigurationsparameter festlegen oder ändern, die nicht im Motif-Verwaltungsprogramm angezeigt werden. Bei dieser Konfigurationsmethode müssen Sie jedoch mehr Informationen angeben als im Motif-Verwaltungsprogramm. Außerdem müssen Sie sicherstellen, dass die von Ihnen angegebenen Informationen gültig und mit vorhandenen Ressourcendefinitionen konsistent sind. (Das Motif-Verwaltungsprogramm wird zuletzt deshalb empfohlen, weil es die Konsistenz der eingegebenen Daten gewährleistet. Darüber hinaus kann es viele Konfigurationswerte von den in Menüs und Dialoganzeigen ausgewählten Optionen ableiten und Werte ausgehend von verfügbaren Definitionen eintragen.)

Die meisten Verwaltungsbefehle werden im Befehlszeilenverwaltungsprogramm **snaadmin** ausgeführt. Sie können **snaadmin**-Befehle im folgenden Format absetzen:

```
snaadmin Befehl, Parameter1=Wert1, Parameter2=Wert2, ...,
           {Subeintrag1}, Subparameter1=Subwert1,
           Subparameter2=Subwert2...
```

Hilfe zum Befehlszeilenverwaltungsprogramm **snaadmin** können Sie mit den folgenden Befehlen aufrufen:

- **snaadmin -h** stellt Basisinformationen zum Befehlszeilenverwaltungsprogramm und Hilfetexte zur Verwendung dieses Programms bereit.
- **snaadmin -h -d** ruft eine Liste von Befehlen auf, die das Programm **snaadmin** ausführen kann.
- **snaadmin -h *Befehl*** ruft einen Hilfetext für den benannten *Befehl* auf.
- **snaadmin -h -d *Befehl*** stellt ausführliche Hilfeinformationen zum benannten *Befehl* und eine Liste der für diesen Befehl verfügbaren Konfigurationsparameter bereit.

Einige Befehle können von einem IBM Remote API Client abgesetzt werden, sofern der Befehl die Option **-n** für die Angabe eines Servernamens enthält. Ein solcher Befehl hat denselben Effekt wie ein vom benannten Server abgesetzter Befehl.

Im verbleibenden Teil dieses Abschnitts sind die Verwaltungsbefehle für verschiedene Arten von Ressourcen zusammengefasst. Einige der Befehlstypen sind wie folgt aufgeführt:

**status\_\***

Liefert eine Übersicht über die Ressourcenarten

**define\_\***

Erstellt einen neuen Eintrag **define\_\*** in der Konfigurationsdatei oder trägt für einen vorhandenen Ressourceneintrag die neue Definition ein

**delete\_\***

Löscht den entsprechenden Eintrag **define\_\*** aus der Datei

## Befehlszeilenverwaltungsprogramm verwenden

- query\_\*** Gibt Informationen aus der Konfigurationsdatei zur entsprechenden Komponente aus, ohne die Datei zu ändern
- set\_\*** Steuert Verwaltungsfunktionen, z. B. Trace-Parameter und Protokollierungsparameter

Ausführliche Informationen zur Befehlszeilenkonfiguration können Sie der Veröffentlichung *Communications Server for Linux Administration Command Reference* entnehmen.

---

## Kapitel 3. Grundlegende Konfigurationsaufgaben

Dieses Kapitel gibt einen Überblick über die Konfigurationsaufgaben und erläutert die Vorgehensweise beim Konfigurieren des CS-Linux-Knotens. Es erläutert außerdem das Konfigurieren von Haupt- und Ausweichservern, wenn Communications Server für Linux in einer Client-Server-Umgebung verwendet wird.

---

### Client-Server-Funktionen konfigurieren

Dieser Abschnitt ist für Sie nur von Relevanz, wenn Sie Communications Server für Linux für die Ausführung in einer Client-Server-Umgebung (mit mehreren CS-Linux-Knoten in einem Netz) installiert haben.

Viele Ressourcen wie Ports und LUs werden auf einem individuellen Knoten konfiguriert. Diese Ressourcen werden als Knotenressourcen bezeichnet.

Andere Ressourcen gelten für alle Knoten gemeinsam. Für diese Ressourcen wird nur eine Definition für die gesamte Domäne gespeichert. Diese Ressourcen werden als Domänenressourcen bezeichnet. Definitionen von Domänenressourcen werden nur auf dem Hauptserver der Domäne gespeichert und sind für alle Knoten in der Domäne zugänglich.

**Anmerkung:** Ein eigenständiges CS-Linux-System hat nur einen Server, der immer die Rolle des Hauptservers übernimmt.

Ein einer Client-Server-Umgebung kann ein Server als Konfigurationsserver markiert werden. Communications Server für Linux verwaltet eine Liste dieser Konfigurationsserver. Der erste aufgelistete Server ist der Hauptserver. Alle weiteren Server in der Liste sind Ausweichserver. Die Server werden der Reihenfolge nach aufgelistet, so dass der zweite aufgelistete Server (der erste Ausweichserver) die Aufgaben des Hauptservers übernimmt, wenn dieser nicht verfügbar ist. Der dritte Server in der Liste (der zweite Ausweichserver) übernimmt diese Aufgaben, falls weder der Hauptserver noch der erste Ausweichserver verfügbar ist, usw.

Wenn Knoten in der Domäne aktiv sind, wird der erste verfügbare Konfigurationsserver der Domäne (der erste Server mit der Software Communications Server für Linux, zu dem eine Verbindung hergestellt werden kann) zum Hauptserver. Falls der aktuelle Hauptserver nicht mehr verfügbar ist (weil keine Verbindung hergestellt wird, weil ein Netzfehler vorliegt oder weil die auf dem Server ausgeführte SNA-Software gestoppt wurde), wird der nächste verfügbare Konfigurationsserver in der Liste zum neuen Hauptserver.

Communications Server für Linux kann ohne einen Hauptserver ausgeführt werden. Dies geschieht, wenn es nicht möglich ist, zu einem der Server in der Liste der Konfigurationsserver eine Verbindung herzustellen. In einer solchen Situation können Sie die Knotenressourcen nur auf den Servern anzeigen und konfigurieren, zu denen eine Verbindung möglich ist.

**Anmerkung:** Sie können nicht direkt angeben, welcher Knoten die Rolle des Hauptservers übernehmen soll. Der Hauptserver wird auf der Basis der Reihenfolge ausgewählt, in der die Knoten zur Liste der Konfigurationsserver hinzugefügt werden.

## Client-Server-Funktionen konfigurieren

Falls Sie einen Server an den Anfang der Liste verschieben möchten, entfernen Sie alle anderen Knoten aus der Liste und fügen Sie sie dann erneut hinzu.

Sie können auch die folgenden Verwaltungsbefehle verwenden, um Konfigurationsserver abzufragen, hinzuzufügen und zu löschen:

### **query\_sna\_net**

Listet die Server in der Datei auf

### **add\_backup**

Fügt am Ende der Liste einen neuen Server hinzu

### **delete\_backup**

Entfernt einen Server aus der Liste. Mit dem Befehl **delete\_backup** können Sie den Hauptserver löschen (so dass der zweite aufgelistete Server die Aufgaben des Hauptservers übernimmt) oder einen Ausweichserver (so dass dieser nicht mehr als Hauptserver eingesetzt werden kann).

**Anmerkung:** Sie können einen Server nicht löschen, wenn er der einzige aufgelistete Server ist, auf dem die Software Communications Server für Linux ausgeführt wird, weil in diesem Fall kein Server mehr verfügbar wäre, der die Aufgaben des Hauptservers übernehmen könnte. In einer Client-Server-Konfiguration ist mindestens ein aktivierter Hauptserver erforderlich.

Kapitel 10, „Client-Server-Systeme mit Communications Server für Linux verwalten“, auf Seite 137 enthält Hinweise zur erweiterten Client-Server-Konfiguration, z. B. zum Verschieben von Clients und Servern in andere CS-Linux-Domänen und zum detaillierten Konfigurieren des Clientbetriebs.

---

## Knoten konfigurieren

Der erste Schritt der CS-Linux-Konfiguration auf einem System ist das Konfigurieren des lokalen Knotens. Die Knotenkonfiguration stellt die Basisdaten bereit, die der Knoten benötigt, um Teil eines APPN-Netzes sein zu können. Erst nachdem Sie den Knoten konfiguriert haben, können Sie die Konnektivität oder andere Ressourcen für den Knoten definieren.

Falls der Knoten bereits konfiguriert wurde, müssen Sie den Knoten stoppen, bevor Sie die Knotenkonfiguration ändern.

Verwenden Sie für das Konfigurieren des Knotens eine der folgenden Methoden:

### **Motif-Verwaltungsprogramm**

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **Knotenparameter konfigurieren** aus.

### **Befehlszeilenverwaltungsprogramm**

Setzen Sie den Befehl **define\_node** ab.

Mit den erweiterten Parametern für die Knotenkonfiguration können Sie Sitzungen mit nicht definierten Partner-LUs, das Melden von Sicherheitsfehlern und Zeitlimits für nutzungsabhängige Ressourcen steuern.

## Parameter für die Knotenkonfiguration

Für die Knotenkonfiguration benötigen Sie die folgenden Informationen:

### *APPN-Unterstützung*

Stufe der APPN-Unterstützung für den Knoten:

- Wenn Ihr Netz ein APPN-Netz ist, konfigurieren Sie den Knoten als LEN-Knoten.
- Wenn der Knoten Teil eines APPN-Netzes sein soll, in dem ein anderer Knoten Dienste für das Sitzungs-Routing bereitstellt, oder DLUR nur auf dem lokalen Knoten verwendet werden soll, konfigurieren Sie den Knoten als Endknoten.
- Soll der Knoten Dienste für die Zwischenpfadsteuerung in einem APPN-Netz oder Durchgriff-DLUR-Dienste für untergeordnete Knoten bereitstellen, konfigurieren Sie den Knoten als Netzknoten.
- Wenn der Knoten Sitzungs-Routing-Dienste für andere Knoten in einem Zweignetz, die nicht Teil des zentralen APPN-Hauptnetzes sind, bereitstellen soll, konfigurieren Sie den Knoten als Zweignetzknoten.

### *Steuerpunktname*

Vollständig qualifizierter Steuerpunktname für den lokalen Knoten. Da dieser Name möglicherweise auch auf anderen Knoten im Netz konfiguriert werden muss, sollten Sie den Namen mit dem zuständigen Planungsbeauftragten für das SNA-Netz abstimmen.

Wenn Sie den Steuerpunkt definieren, definiert Communications Server für Linux automatisch eine LU, die denselben Namen wie der Steuerpunkt hat. Diese LU kann für den Knoten als lokale Standard-LU verwendet werden.

### *Aliasname für Steuerpunkt*

Lokaler Aliasname für die lokale Standard-LU. Geben Sie diesen Wert an, wenn die lokale Standard-LU von unabhängigen LUs des Typs 6.2 verwendet wird.

### *Knoten-ID*

Kennung der PU auf dem lokalen Knoten. Geben Sie nur dann einen Wert an, wenn der Knoten für abhängigen Verkehr unter Verwendung der Standard-LU (Steuerpunkt-LU) eingesetzt wird.

## Zusätzliche Konfiguration

Führen Sie nach dem Konfigurieren des Knotens die folgenden Konfigurationsaufgaben aus:

- Konfigurieren Sie die Konnektivität wie in Kapitel 4, „Konnektivitätskomponenten definieren“, auf Seite 63 beschrieben.
- Konfigurieren Sie Knotenressourcen (LUs) wie in Kapitel 6, „APPC-Kommunikation konfigurieren“, auf Seite 85 oder Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 beschrieben.
- Konfigurieren Sie Anwendungen wie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 beschrieben.

### Protokollierung konfigurieren

Communications Server für Linux schreibt Protokollnachrichten zu Ereignissen, die von der Norm abweichen, und optional auch zu normalen Ereignissen in Protokolldateien. Wenn Sie versuchen, einen Fehler zu bestimmen, sollten Sie zunächst die Protokolldateien aufrufen, denn die Protokollnachrichten geben Hinweise auf die Fehlerursache und auf erforderliche Maßnahmen.

Communications Server für Linux protokolliert Nachrichten für die folgenden Ereigniskategorien:

**Fehler** Ein anomales Ereignis, das den Systembetrieb in für den Benutzer spürbarer Weise beeinträchtigt (z. B. die anomale Beendigung einer Sitzung).

**Ausnahme**

Ein anomales Ereignis, das den Systembetrieb in für den Benutzer nicht unmittelbar spürbarer Weise beeinträchtigt (z. B. ein Mangel an Ressourcen), oder ein Ereignis, das den Systembetrieb nicht beeinträchtigt, aber auf die Ursache späterer Ausnahmen oder Fehler hinweisen kann (z. B. der Empfang einer unerwarteten Nachricht vom fernen System).

**Prüfung**

Ein normales Ereignis (z. B. das Starten einer Sitzung).

Communications Server für Linux verwaltet außerdem eine Auslastungsprotokolldatei, in der Informationen zur aktuellen und maximalen Nutzung der CS-Linux-Ressourcen erfasst werden.

Damit die Protokolle für normale Bedingungen und Fehlerbedingungen besser unterschieden werden können, werden Nachrichten unterschiedlicher Kategorien in gesonderten Dateien protokolliert. Fehler- und Ausnahmenachrichten werden in der Fehlerprotokolldatei und Prüfnachrichten in der Prüfprotokolldatei protokolliert.

Communications Server für Linux stellt einen Sicherungsmechanismus bereit, der verhindert, dass die Protokolldateien zu groß werden und Plattenressourcen binden. Erreicht eine Protokolldatei die maximal zulässige Größe, kopiert Communications Server für Linux den aktuellen Inhalt dieser Datei in eine Sicherungsdatei und löscht die Protokolldatei.

Communications Server für Linux verwendet standardmäßig die folgenden Protokolldateien:

**Fehlerprotokolldatei**

`/var/opt/ibm/sna/sna.err`

`/var/opt/ibm/sna/bak.err` (Sicherung)

**Prüfprotokolldatei**

`/var/opt/ibm/sna/sna.aud`

`/var/opt/ibm/sna/bak.aud` (Sicherung)

**Nutzungsprotokolldatei**

`/var/opt/ibm/sna/sna.usage`

`/var/opt/ibm/sna/bak.usage` (Sicherung)

Sie können die Protokolldateien wie folgt mit einem Texteditor oder anderen Linux-Systemdienstprogrammen anzeigen:

- vi** Zeigt den Text der Datei an. Im Editor können Sie in der Datei vor- und zurückblättern und nach bestimmten Einträgen suchen.
- pg** Zeigt eine Datei seitenweise an. Dieses Dienstprogramm ist einfach anzuwenden, jedoch nur für kleine Protokolldateien sinnvoll.
- tail** Anzeigen des Nachsatzes (Endes) einer Datei. Am Ende der Datei stehen die aktuellsten Protokollnachrichten. Wenn Sie dieses Dienstprogramm mit der Option **-f** verwenden, können Sie die Protokolldatei bei laufendem Systembetrieb überwachen.

Falls Sie statt der ausführlichen die kurze Protokollierung ausgewählt haben, können Sie mit dem Befehl **snahelp** für eine bestimmte Nachrichtennummer Informationen zu Ursache und erforderlichen Maßnahmen aufrufen.

Für die meisten Situationen sind die Standardeinstellungen für die Protokollierung ausreichend. Sie können jedoch die folgenden Änderungen vornehmen:

- Sie können angeben, welche Nachrichtenkategorien protokolliert werden sollen. Fehlernachrichten werden immer protokolliert und können nicht inaktiviert werden. Für die beiden anderen Nachrichtenkategorien ist die Protokollierung normalerweise inaktiviert. Sie können sie bei Bedarf jedoch aktivieren.
- Sie können die Ausführlichkeit von Protokollnachrichten bestimmen.
- Sie können die zentrale Protokollierung für die Domäne oder die lokale Protokollierung für Einzelknoten angeben.
- Sie können Namen und Größe der Protokolldateien ändern.

Verwenden Sie für das Konfigurieren der Protokollierung eine der folgenden Methoden:

### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Diagnoseprogramm** des Knotenfensters oder des Domänenfensters den Eintrag **Protokollierung** aus.

### Befehlszeilenverwaltungsprogramm

Setzen Sie einen der folgenden Befehle ab:

- **set\_central\_logging**
- **set\_global\_log\_type**
- **set\_log\_type**
- **set\_log\_file**

Der Protokollierungsdialog im Motif-Verwaltungsprogramm wirkt sich auf die Protokolleinstellungen in der gesamten Domäne aus. In der Befehlszeile können Sie die Domäneneinstellungen außer Kraft setzen, indem Sie lokale Protokolleinstellungen für eine bestimmte Maschine konfigurieren.

Im Motif-Verwaltungsprogramm können Sie nicht nur die Protokollierung, sondern auch den Trace auf Knotenebene steuern. Die Befehlszeilenschnittstelle bietet noch mehr Funktionen für die Steuerung von Protokollierung und Trace-Durchführung an. Weitere Informationen zur Protokollierung und zum Durchführen von Traces finden Sie im *Communications Server for Linux Diagnostics Guide*.



---

## Kapitel 4. Konnektivitätskomponenten definieren

Voraussetzung für die Kommunikation des CS-Linux-Knotens mit anderen Knoten ist das Konfigurieren der Konnektivität zu mindestens einem Nachbarknoten. Eine Verbindung kann für abhängigen Verkehr, unabhängigen Verkehr oder beide Arten von Verkehr konfiguriert werden.

In Ihrem Computer können Adapter für ein oder mehrere Verbindungsprotokoll(e) installiert sein. Viele der Informationen, die Sie beim Konfigurieren der Konnektivität eingeben müssen, hängen vom verwendeten Verbindungsprotokoll ab. Der ferne Knoten muss ebenfalls mit einer Adapterkarte des von Ihnen ausgewählten Typs ausgestattet sein. Ist dies nicht der Fall, muss sich zwischen dem lokalen und dem fernen Knoten eine Brücke oder ein Router befinden. Eine Liste der von Communications Server für Linux unterstützten Verbindungsprotokolle finden Sie im Abschnitt „DLCs, Ports und Verbindungsnetze definieren“ auf Seite 64.

Zum Konfigurieren einer Verbindung müssen Sie einen Port definieren. Gehen Sie dabei wie im Abschnitt „DLCs, Ports und Verbindungsnetze definieren“ auf Seite 64 beschrieben vor. In den meisten Fällen müssen Sie außerdem eine Verbindungsstation konfigurieren. Anweisungen hierfür finden Sie im Abschnitt „Verbindungsstationen definieren“ auf Seite 69. Wenn LUs auf dem lokalen Knoten über DLUR mit einem Host kommunizieren sollen, müssen Sie auf dem lokalen Knoten eine DLUR-PU definieren. Gehen Sie dabei wie im Abschnitt „DLUR-PUs definieren“ auf Seite 77 beschrieben vor.

Wenn Sie das Motif-Verwaltungsprogramm verwenden, wird im Rahmen der Port-Konfiguration automatisch eine Datenübertragungssteuerung (DLC) konfiguriert. Sie können den Port aber auch als Teil eines Verbindungsnetzes definieren. Wenn Sie das Befehlszeilenverwaltungsprogramm verwenden, muss dieser Konfigurationsschritt unabhängig von der Port-Konfiguration ausgeführt werden.

Die für das Konfigurieren einer Verbindung erforderlichen Informationen richten sich nach dem Verbindungsprotokoll und danach, ob Ihr Netz ein APPN-Netz ist und ob die Verbindung für abhängigen Verkehr, unabhängigen Verkehr oder für beide Arten von Verkehr konfiguriert werden soll. Welche Verbindungen Sie konfigurieren müssen, hängt außerdem davon ab, welche Art der Kommunikation unterstützt werden soll:

**LUA** Wenn Sie mit LUA arbeiten wollen, müssen Sie eine Verbindung zum Host konfigurieren. Die Verbindung muss für abhängigen Verkehr konfiguriert werden. Da die Verbindung sowohl auf dem Host als auch auf dem CS-Linux-Knoten konfiguriert werden muss, sollten Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz wenden.

### **Verwendung von CPI-C oder APPC**

Wenn Sie mit CPI-C oder APPC arbeiten wollen und Ihr Netz ein APPN-Netz ist, müssen Sie Verbindungen zu allen Nachbarknoten, auf die der Zugriff möglich sein soll, konfigurieren. Diese Verbindungen müssen für unabhängigen Verkehr konfiguriert werden. Die Konfiguration ist sowohl auf den Nachbarknoten als auch auf dem CS-Linux-Knoten auszuführen, so dass Sie sich unter Umständen an den zuständigen Planungsbeauftragten für das SNA-Netz wenden müssen.

## Konnektivitätskomponenten definieren

### Verwendung als APPN-Knoten

Wenn der CS-Linux-Knoten ein Endknoten oder Netzknoten in einem APPN-Netz ist, kann die Zahl der von Ihnen zu konfigurierenden Verbindungen stark reduziert werden. Sie können Verbindungen zu einem oder mehreren benachbarten Netzknoten konfigurieren und über diese Verbindungen auf alle Knoten des APPN-Netzes zugreifen. Wenn Sie auf andere Nachbarknoten direkt zugreifen wollen, können Sie zu diesen ebenfalls Verbindungen konfigurieren. Dies ist in der Regel nicht nötig, kann jedoch den Durchsatz erhöhen. Wenn die Nachbarknoten über Token-Ring miteinander verbunden sind, können Direktverbindungen dynamisch hergestellt werden und müssen nicht konfiguriert werden. Sie müssen lediglich sicherstellen, dass Sie das Netz beim Definieren des Ports als ein Verbindungsnetz konfigurieren.

Von unabhängigem APPC können die Vorzüge des APPN-Netzbetriebs immer genutzt werden. Für LUA ist dies nur über die DLUR-Funktion möglich. (DLUR unterstützt die Kommunikation zwischen einem Host und abhängigen LUs auf dem lokalen Knoten oder auf untergeordneten Knoten in einem APPN-Netz.) Sie können DLUR nur verwenden, wenn Ihr Host DLUS unterstützt. Sollten Sie an der Anwendung von DLUR interessiert sein, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

---

## DLCs, Ports und Verbindungsnetze definieren

Ein Port ist das lokale Ende einer DFV-Verbindung und ein eindeutiger Zugriffspunkt im Netz. Jeder Port ist einem bestimmten Verbindungsprotokoll zugeordnet. Dabei kann es sich um eines der folgenden Verbindungsprotokolle handeln:

- SDLC
- Token-Ring
- Ethernet
- X.25 oder QLLC (qualifizierte Steuerung logischer Verbindungen)
- MultiPath Channel (MPC) (nur Communications Server für Linux auf System z)
- Enterprise Extender (HPR/IP)

Für ein bestimmtes Verbindungsprotokoll können Sie mehr als einen Port konfigurieren. Im Allgemeinen ist ein Port ein physischer Zugriffspunkt (z. B. eine Adapterkarte). Einige Verbindungsprotokolle (z. B. Token-Ring) gestatten Ihnen allerdings, für einen Adapter mehrere Ports zu definieren. Die verschiedenen Ports werden durch Adressen (z. B. die SAP-Nummer) unterschieden.

Wenn Sie einen Port für ein bestimmtes Verbindungsprotokoll im Motif-Verwaltungsprogramm definieren, definiert Communications Server für Linux für diesen Port automatisch eine DLC, sofern noch keine DLC dieses Typs definiert wurde. Im Befehlszeilenverwaltungsprogramm müssen Sie den Port und die DLC mit verschiedenen Befehlen definieren.

In einem APPN-Netz mit Token-Ring-Verbindungsprotokollen können Sie im Konfigurationsdialog für SAPs angeben, dass der Port Teil eines Verbindungsnetzes ist.

Wenn Sie SNA Gateway verwenden, können Sie eine Schablone definieren, mit der Definitionen für implizite Verbindungsstationen (nicht explizit konfigurierte Verbindungsstationen) generiert werden. Implizite Verbindungsstationen können untergeordnete LUs unterstützen. Werden Felder für implizite PUs geändert, wäh-

rend der Port aktiv ist, wirken sich die Änderungen auf alle nach der Änderung generierten Instanzen impliziter Verbindungsstationen aus.

Verwenden Sie zum Konfigurieren eines Ports, eines Verbindungsnetzes und einer DLC eine der folgenden Methoden:

### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **Konnektivität** und dann die Option **Neuer Port** aus.

### Befehlszeilenverwaltungsprogramm

Konfigurieren einer DLC:

```
define_Typ_dlc
```

Konfigurieren eines Ports:

```
define_Typ_port
```

In diesen Befehlen steht *Typ* für den Typ des Verbindungsprotokolls (sdlc, tr, ethernet, qllc, mpc, ip).

Konfigurieren eines Verbindungsnetzes:

```
define_cn
```

Mit den erweiterten Port-Konfigurationsparametern können Sie die BTU-Größe, die zulässige Anzahl aktiver Verbindungen, das Generieren impliziter untergeordneter LUs und die Einstellungen für dynamische Verbindungsstationen steuern.

## Konfigurationsparameter für DLCs, Verbindungsnetze und Ports

Für die Port-Konfiguration sind die folgenden Parameter erforderlich. (Wenn Sie das Motif-Verwaltungsprogramm verwenden, können Sie beim Konfigurieren eines Ports Informationen zur DLC bereitstellen und den Port einem Verbindungsnetz zuordnen.)

### *SNA-Port-Name*

Der lokal bekannte Name des Ports.

### *Adapterkartennummer*

Dieses Feld wird nicht für Enterprise-Extender-Ports verwendet.

Eine Nummer, die die zu verwendende Adapterkarte angibt, sofern in diesem Computer mehr als eine Karte desselben Typs installiert ist.

### *Port-Nummer*

Dieses Feld wird nicht für Enterprise-Extender-Ports verwendet.

Die Nummer des zu verwendenden Ports, sofern die Adapterkarte mehr als einen Port unterstützen kann. Der Bereich gültiger Port-Nummern beginnt bei 0 und endet bei der Differenz aus der Zahl vom Adapter unterstützter Ports und eins. Geben Sie für den ersten Port der Adapterkarte 0 ein.

Dieses Feld findet nur Anwendung, wenn die Adapterkarte mehr als einen Port unterstützen kann.

### *Aktiv bei Initialisierung*

Information, ob der Port beim Start des Knotens automatisch aktiviert wer-

## DLCs, Ports und Verbindungsnetze definieren

den soll. Diese Einstellung ermöglicht das Aktivieren von Verbindungsstationen, die bei Anforderungen von Nachbarknoten oder des lokalen Knotens aktiviert werden können, wenn der Port aktiv ist. (Die Verbindungsstationen werden nicht zusammen mit dem Port, sondern separat aktiviert.)

In den folgenden Abschnitten sind die zusätzlichen Port-Parameter beschrieben, die für die jeweilige Verbindungsart spezifisch sind. Für QLLC sind keine weiteren Port-Parameter erforderlich.

### Zusätzliche Port-Parameter für SDLC

#### *Leitungsangaben*

Die folgenden Parameter beschreiben die Art der SDLC-Verbindung:

*Typ* Wählen Sie einen der folgenden Werte aus:

#### **Standleitung**

Für die SDLC-Verbindung zwischen diesem Computer und dem fernen System wird eine dedizierte Leitung verwendet.

#### **Ankommende Wählleitung**

Das Standardtelefonnetz wird für ankommende Anrufe verwendet.

Für einen Port, der (gemäß Angabe im Feld *Funktion der Verbindungsstation*) kein primärer Port ist, müssen Sie außerdem die Aufrufadresse konfigurieren. (Für abgehende Anrufe wird diese Adresse auf der Verbindungsstation konfiguriert.) Die Aufrufadresse ist eine Einzelbyteadresse (Standardwert ist C1), die mit der auf der fernen Verbindungsstation konfigurierten Aufrufadresse übereinstimmen muss. Wenn der Port aktiv ist, antwortet er auf Rahmen, die mit dieser Aufrufadresse gesendet werden.

Für einen primären Port müssen Sie keine Aufrufadresse konfigurieren. Der Port verwendet die von der fernen Verbindungsstation im ankommenden Anruf angegebene Aufrufadresse. Für andere Port-Typen wird die Aufrufadresse auf den einzelnen Verbindungsstationen konfiguriert.

#### **Abgehende Wählleitung**

Das Standardtelefonnetz wird für abgehende Anrufe verwendet.

#### *Funktion der Verbindungsstation*

Wählen Sie einen Wert aus, der die Funktion des lokalen Knotens für Verbindungsstationen beschreibt, die für diesen Port definiert sind. In der SDLC-Kommunikation übernimmt ein Endpunkt die Verwaltung der Verbindung und wird als primäre Verbindungsstation bezeichnet. Der andere Endpunkt ist die sekundäre Verbindungsstation.

Verwenden Sie für dieses Feld einen der folgenden Werte:

#### **Sekundär**

Das andere Ende der Verbindung soll der Controller sein, und das ferne System ist als Primärsystem konfiguriert.

## DLCs, Ports und Verbindungsnetze definieren

Dies ist fast immer der Fall, wenn Sie eine Verbindung zu einem Hostsystem konfigurieren.

**Primär** Dieser Port soll als SDLC-Controller der Verbindung fungieren, und das ferne System soll als Sekundärsystem konfiguriert werden.

### Aushandelbar

Diese Einstellung ermöglicht beiden Enden ganz flexibel auszuhandeln, welches System die Funktion des Primärsystems übernehmen soll. Wählen Sie diesen Wert aus, wenn Sie nicht wissen, welche Funktion für das ferne System konfiguriert ist.

Diese Einstellung können Sie für eine Peerverbindung verwenden. Sie müssen jedoch bedenken, dass das Aushandeln der Funktion beim Aktivieren der Verbindung zu einer kurzen Verzögerung führt.

### Primär für Mehrpunktverbindung

Die Verbindung wird über eine Standleitung hergestellt, und dieser Port soll als Controller einer Mehrpunktverbindung zu verschiedenen sekundären Knoten eingesetzt werden.

Verwenden Sie diese Einstellung, wenn Sie mehrere Verbindungsstationen vom lokalen Knoten zu verschiedenen fernen Knoten (z. B. für Verbindungen zu untergeordneten Knoten) konfigurieren möchten. Jeder dieser fernen Knoten muss als sekundärer Knoten konfiguriert werden, und Sie müssen eine Standleitung benutzen.

### Sekundär, mehrere PUs

Der lokale Port ist eine der sekundären Stationen einer Mehrpunktverbindung, die vom Port des fernen Systems gesteuert wird.

Bei Fragen zur Konfiguration dieser Parameter sollten Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz wenden.

## Zusätzliche Port-Parameter für Token-Ring und Ethernet

### *Lokale SAP-Nummer*

Die Adresse des SAP lautet für Intel- und OSA2-Adapter in der Regel 04. Verwenden Sie einen anderen Wert, falls Sie mehr als einen SAP der Karte verwenden. Für einen OSA-Express-Adapter muss die lokale SAP-Nummer mit der Nummer übereinstimmen, die in der OSA/SF für die E/A-Einheitenadressen definiert wurde, die der ethX-Schnittstelle dieses Linux-Images entsprechen.

Die SAP-Nummer muss ein Vielfaches von 4 sein.

Sollten Sie nicht wissen, welchen Wert Sie in diesem Feld eingeben müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

### *Auf Verbindungsnetz definieren*

Information, ob der SAP auf das LAN als ein Verbindungsnetz zugreifen soll. Bei Definition eines Verbindungsnetzes können Verbindungen zwischen Knoten des Verbindungsnetzes ohne vorherige Konfiguration dynamisch aktiviert werden.

## DLCs, Ports und Verbindungsnetze definieren

Dieses Feld findet nur Anwendung, wenn der lokale Knoten kein LEN-Knoten ist, denn LEN-Knoten können keine Verbindungsnetze verwenden.

### *Name des Verbindungsnetzes*

Der Name des Verbindungsnetzes. Sie müssen den Namen des Verbindungsnetzes nur angeben, wenn Sie die Option *Auf Verbindungsnetz definieren* angegeben haben, um den SAP für ein Verbindungsnetz zu definieren. Der Name des Verbindungsnetzes wird als Name des virtuellen Routing-Knotens verwendet, um zwischen den Knoten des Verbindungsnetzes Verbindungen herzustellen.

Geben Sie auf allen Knoten des Verbindungsnetzes denselben Namen des Verbindungsnetzes an.

### *Ethernet-Typ*

Dieses Feld findet nur bei Ethernet-Verbindungen Anwendung.

Gibt an, ob das Netz ein Standard-Ethernet-Netz oder ein IEEE-802.3-Netz ist.

## **Zusätzliche Parameter für Enterprise Extender (HPR/IP)**

### *Lokale IP-Schnittstelle*

Dies ist ein optionales Feld, in dem Sie die lokale IP-Netzschnittstelle angeben können, die für den Port verwendet werden soll, wenn Sie Zugriff auf mehrere IP-Netze haben. Falls Sie nur Zugriff auf ein IP-Netz haben, können Sie dieses Feld leer lassen.

Wenn Sie die Schnittstelle angeben müssen, können Sie eine der folgenden Angaben verwenden:

- IPV4-Adresse in Schreibweise mit Trennzeichen (z. B. 193.1.11.100)
- IPV6-Hexadezimaladresse mit Doppelpunkt (z. B. 2001:0db8:0000:0000:0000:1428:57ab oder 2001:db8::1428:57ab)
- Schnittstellenkennung (z. B. eth0 oder en0)

### *Protokoll*

Gibt an, ob Verbindungen an diesem Port IPV4- oder IPV6-Adressen verwenden

### *Auf Verbindungsnetz definieren*

Information, ob die Adapterkarte auf das LAN als ein Verbindungsnetz zugreifen soll. Bei Definition eines Verbindungsnetzes können Verbindungen zwischen Knoten des Verbindungsnetzes ohne vorherige Konfiguration dynamisch aktiviert werden.

Dieses Feld findet nur Anwendung, wenn der lokale Knoten kein LEN-Knoten ist, denn LEN-Knoten können keine Verbindungsnetze verwenden.

### *Name des Verbindungsnetzes*

Der Name des Verbindungsnetzes. Sie müssen den Namen des Verbindungsnetzes nur angeben, wenn Sie die Option *Auf Verbindungsnetz definieren* angegeben haben, um den Port für ein Verbindungsnetz zu definieren. Der Name des Verbindungsnetzes wird als Name des virtuellen Routing-Knotens verwendet, um zwischen den Knoten des Verbindungsnetzes Verbindungen herzustellen.

Geben Sie auf allen Knoten des Verbindungsnetzes denselben Namen des Verbindungsnetzes an.

### Zusätzliche Port-Parameter für implizite Verbindungen

#### *Max. Anzahl aktiver Schabloneninstanzen*

Geben Sie die maximale Anzahl der Verbindungsstationsinstanzen an, die von der Schablone generiert werden sollen.

#### *Untergeordnete LUs für Zugriff auf implizite PUs konfigurieren*

Information, ob untergeordnete LUs konfiguriert werden sollen, die diese PU verwenden (siehe Abschnitt „SNA Gateway konfigurieren“ auf Seite 122).

#### *HPR auf impliziten Verbindungen unterstützt*

Information, ob das High-Performance Routing auf impliziten Verbindungsstationen unterstützt wird.

#### *Fehlerbehebung auf Verbindungsebene für implizite Verbindungen*

Information, ob der HPR-Verkehr auf impliziten Verbindungen unter Verwendung der Fehlerbehebung auf Verbindungsebene gesendet werden soll.

### Zusätzliche Konfiguration

Führen Sie nach der Port-Konfiguration die folgenden Konfigurationsaufgaben aus:

- Wie Sie für einen von Ihnen konfigurierten Port eine Verbindungsstation definieren, können Sie im Abschnitt „Verbindungsstationen definieren“ nachlesen.
- Wie Sie eine DLUR-PU definieren, können Sie im Abschnitt „DLUR-PUs definieren“ auf Seite 77 nachlesen.
- Wie Sie Unterstützung für die APPC-Kommunikation definieren, können Sie in Kapitel 6, „APPC-Kommunikation konfigurieren“, auf Seite 85 nachlesen.

---

## Verbindungsstationen definieren

Für die Kommunikation mit anderen Knoten in einem SNA-Netz müssen Sie die Kenndaten einer Verbindungsstation zu einem Nachbarknoten im SNA-Netz konfigurieren. Bevor Sie eine Verbindungsstation definieren, müssen Sie für den von Ihnen verwendeten Adapter (und das verwendete Verbindungsprotokoll) einen Port definieren. Die meisten für das Konfigurieren einer Verbindungsstation benötigten Informationen sind für alle Protokolle identisch.

Eine Verbindungsstation ist der logische, über das SNA-Netz führende Pfad zwischen dem lokalen CS-Linux-Knoten und einem fernen Computer. Bei dem fernen Computer kann es sich um einen der folgenden Computer handeln:

- Ein Hostcomputer, auf dem Communications Server für Linux mit Hilfe der 3270- oder der LUA-Kommunikation auf ein Hostprogramm zugreift (oder der APPC bzw. CPI-C für die DFV zwischen Programmen verwendet)
- Ein Peercomputer, der über Communications Server für Linux gleichberechtigt mit einem fernen Computer kommuniziert (die typische Aufteilung in einem APPN-Netz)
- Ein untergeordneter Computer, der das CS-Linux-Feature SNA Gateway oder DLUR für den Zugriff auf einen Host verwendet.

Eine Verbindungsstation ist einem bestimmten Port zugeordnet. Sie können für jeden Port eine oder mehrere Verbindungsstationen definieren.

## Verbindungsstationen definieren

Jeder Verbindungsstation, die abhängigen Verkehr unterstützt, ist eine physische Einheit (PU, Physical Unit) zugeordnet. Aufgrund der Zuordnung zwischen PUs und Verbindungsstationen behandelt Communications Server für Linux die physischen Einheiten nicht wie gesonderte Ressourcen. Sie werden stattdessen im Rahmen der Konfiguration der Verbindungsstationen konfiguriert und zusammen mit den Verbindungsstationen gestartet und gestoppt. Verbindungsstationen werden im Konnektivitätsteilfenster des Knotenfensters angezeigt. PUs erscheinen in keinem der Fenster.

**Anmerkung:** In den meisten Situationen müssen Sie zum Port eine Verbindungsstation hinzufügen. Wenn die Verbindung jedoch stets vom fernen Knoten aktiviert werden soll und Sie nur eine dynamisch eingerichtete Verbindungsstation für untergeordneten SNA-Gateway-Verkehr oder für APPC-Verkehr verwenden wollen, müssen Sie nicht explizit eine Verbindungsstation konfigurieren.

Wenn ein ferner Knoten versucht, eine Verbindung zum lokalen Knoten herzustellen, und auf dem lokalen Knoten ein geeigneter Port, aber keine Verbindungsstation definiert ist, die mit der im ankommenden Anruf angegebenen Adresse übereinstimmt, kann Communications Server für Linux eine solche Station implizit definieren. Diese dynamisch eingerichtete Verbindungsstation wird für die Dauer der Verbindung im Konnektivitätsteilfenster des Knotenfensters angezeigt.

Verwenden Sie für das Konfigurieren einer Verbindungsstation eine der folgenden Methoden:

### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **Konnektivität** und dann die Option **Neue Verbindungsstation** aus.

### Befehlszeilenverwaltungsprogramm

Geben Sie den folgenden Befehl ein:

```
define_Typ_ls
```

In diesem Befehl steht *Typ* für den Typ des Verbindungsprotokolls (sd1c, tr, ethernet, ql1c, mpc, ip).

Mit den erweiterten Parametern für Verbindungsstationen können Sie die Übertragungsdaten, den XID-Austausch, optionale Verbindungsfunktionen, die Komprimierung für LU-0-3-Sitzungen, die die Verbindung verwenden, und Reaktivierungsprozeduren steuern.

## Konfigurationsparameter für Verbindungsstationen

Der Konfigurationsdialog für Verbindungsstationen im Motif-Programm enthält die folgenden Abschnitte mit jeweils unterschiedlichen Kategorien von Konfigurationsparametern:

### Verbindungsstation

Geben Sie in diesem Bereich der Dialoganzeige Informationen ein, die für alle Verbindungsstationen erforderlich sind, ganz gleich, ob sie LU-Verkehr für abhängige LUs, unabhängige LUs oder beide Arten von LUs unterstützen. Eine Beschreibung der entsprechenden Parameter finden Sie im Abschnitt „Allgemeine Parameter für Verbindungsstationen“ auf Seite 71.

### Unabhängiger LU-Verkehr

Geben Sie diese Information nur an, wenn Sie die Verbindungsstation für unabhängigen Verkehr verwenden. Eine Beschreibung der entsprechenden Parameter finden Sie im Abschnitt „Parameter für unabhängigen LU-Datenverkehr“ auf Seite 74.

### Abhängiger LU-Verkehr

Geben Sie diese Information nur an, wenn Sie die Verbindungsstation für abhängigen Verkehr verwenden. Eine Beschreibung der entsprechenden Parameter finden Sie im Abschnitt „Parameter für abhängigen LU-Datenverkehr“ auf Seite 75.

## Allgemeine Parameter für Verbindungsstationen

Die folgenden Parameter sind für alle Verbindungsstationen erforderlich, ganz gleich, ob sie abhängigen Verkehr, unabhängigen Verkehr oder beide Arten von Verkehr unterstützen.

Weitere Informationen zu den Parametern in dieser Dialoganzeige finden Sie in den Onlinehilfetexten oder in der Veröffentlichung *Communications Server for Linux Administration Command Reference*.

*Name* Ein Name, über den die Verbindungsstation lokal identifiziert wird.

*SNA-Port-Name*

Der Port, der für den Zugriff auf den Nachbarknoten verwendet wird.

*Aktivierung*

Zum Aktivieren der Verbindungsstation verwendete Methode. Geben Sie eine der folgenden Methoden an:

#### **Durch Administrator**

Die Verbindungsstation wird nur auf Anforderung eines lokalen Systemadministrators aktiviert.

#### **Beim Initialisieren des Knotens**

Die Verbindungsstation wird beim Initialisieren des Knotens automatisch aktiviert.

#### **Bei Anforderung**

Die Verbindungsstation wird automatisch aktiviert, wenn sie für eine Anwendung Konnektivität bereitstellen muss.

Verbindungsstationen werden unabhängig von Ports aktiviert, so dass die Verbindungsstation auch dann aktiviert werden muss, wenn der Port bereits aktiv ist. Allein durch Aktivierung des Ports werden noch keine Verbindungsstationen aktiviert. Ist der Port als 'Aktiv bei Initialisierung' konfiguriert, bedeutet dies nicht, dass Verbindungsstationen bei Initialisierung des Knotens automatisch aktiviert werden. Die Aktivierung eines Ports ist jedoch Voraussetzung für das Aktivieren von Verbindungsstationen. Eine Verbindungsstation kann erst aktiviert werden, wenn der Port auf dem lokalen Knoten und der Port des Nachbarknotens aktiviert sind.

Wenn Sie die Gebühren für die Nutzung der Verbindung übernehmen müssen, sollten Sie die Verbindung nicht unnötig aktivieren, um die Kosten niedrig zu halten.

Sollten Sie nicht sicher sein, auf welchen Wert Sie dieses Feld setzen müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

## Verbindungsstationen definieren

### *LU-Datenverkehr*

Die Art des LU-Datenverkehrs, der über die Verbindung fließen soll. Diese Auswahl bestimmt, welche weiteren Parameter für die Definition der Verbindung erforderlich sind.

Dieser Parameter wird nicht für Enterprise-Extender-Verbindungen (HPR/IP) verwendet, weil diese Verbindungsarten nur unabhängigen Datenverkehr unterstützen.

**Jeder** Die Verbindungsstation kann sowohl für unabhängigen als auch für abhängigen LU-Datenverkehr verwendet werden. Neben den hier genannten Parametern müssen Sie für diese Option die im Abschnitt „Parameter für unabhängigen LU-Datenverkehr“ auf Seite 74 und im Abschnitt „Parameter für abhängigen LU-Datenverkehr“ auf Seite 75 beschriebenen Parameter angeben.

#### **Nur unabhängiger Datenverkehr**

Die Verbindungsstation kann nur für unabhängigen LU-Datenverkehr verwendet werden. Neben den hier genannten Parametern müssen Sie für diese Option die im Abschnitt „Parameter für unabhängigen LU-Datenverkehr“ auf Seite 74 beschriebenen Parameter angeben.

#### **Nur abhängiger Datenverkehr**

Die Verbindungsstation kann nur für abhängigen LU-Datenverkehr verwendet werden. Neben den hier genannten Parametern müssen Sie für diese Option die im Abschnitt „Parameter für abhängigen LU-Datenverkehr“ auf Seite 75 beschriebenen Parameter angeben.

Für die Herstellung einer Verbindung zum Nachbarknoten müssen Sie außerdem Adressierungsinformationen bereitstellen. Die Art der erforderlichen Adressierungsinformationen richtet sich nach dem DLC-Typ des Ports. Wenn Sie keine Adresse für den fernen Knoten angeben, übernimmt die Verbindungsstation die Funktion einer nicht selektiven empfangsbereiten Verbindungsstation und akzeptiert ankommende Anrufe von allen fernen Knoten.

### **Zusätzliche Verbindungsstationsparameter für SDLC:**

#### *Aufrufadresse*

Die Aufrufadresse der fernen Datenstation. Geben Sie die Adresse als zweistelligen Hexadezimalwert (1-Byte-Wert) an. Der Wertebereich beginnt normalerweise bei C1. Eine primäre Verbindungsstation verwendet diesen Wert zum Aufrufen der fernen Datenstation. Eine sekundäre Verbindungsstation antwortet mit diesem Wert auf einen Aufruf. Die Aufrufadresse wird je nach Funktion der Verbindungsstation wie folgt eingegeben:

- Handelt es sich um eine Punkt-zu-Punkt-Verbindung (keine Mehrpunktverbindung), wird normalerweise die Adresse C1 verwendet.
- Wenn der Eltern-Port für diese Verbindung ein Port mit ankommender Wählleitung ist, wird die Aufrufadresse für den Port konfiguriert und kann nicht unabhängig für jede Verbindungsstation konfiguriert werden.
- Wenn Sie eine primäre Verbindungsstation mit abgehender Wählleitung konfigurieren und die Aufrufadresse der fernen Sekundärstation, mit der Sie kommunizieren wollen, nicht kennen, können Sie auf der primären Station die Aufrufadresse 0xFF angeben. Bei dieser Einstellung kann der Knoten Antworten von einer Sekundärstation unabhängig von deren Aufrufadresse empfangen. 0xFF ist für andere als primäre Verbindungen oder Verbindungen über abgehende Wählleitungen keine gültige Adresse.

- In einer Konfiguration mit Mehrpunktverbindung müssen alle sekundären Verbindungsstationen, die mit derselben primären Verbindungsstation kommunizieren, unterschiedliche Aufrufadressen haben.

Die Aufrufadressen an beiden Enden der Verbindung müssen übereinstimmen. Sollten Sie nicht wissen, welche Adresse auf dem fernen System konfiguriert ist, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Auf einem VTAM-Host wird die Aufrufadresse in der VTAM-PU-Definition mit dem Parameter *ADDR=* konfiguriert.

Bei einer AS/400 ist die Aufrufadresse der Parameter *STNADR* der Leitungsbeschreibung.

### Zusätzliche Verbindungsstationsparameter für Token-Ring und Ethernet:

#### *MAC-Adresse*

Die als Folge von Hexadezimalziffern einzugebende MAC-Adresse der fernen Station. Über die MAC-Adresse kann die Adapterkarte des fernen Systems eindeutig identifiziert werden.

Sollten Sie nicht wissen, welchen Wert Sie verwenden müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Ist das ferne Ende dieser Verbindung ein VTAM-Host, entspricht die MAC-Adresse dem Parameter *MACADDR=* der VTAM-Port-Definition.

Wenn Sie eine Verbindung zu einer AS/400 konfigurieren, ist die MAC-Adresse der Parameter *ADPTADR* der Leitungsbeschreibung.

#### *SAP-Nummer*

Die SAP-Nummer des Ports des fernen Computers. Mit der SAP-Nummer können verschiedene Verbindungen, die dieselbe Adapterkarte benutzen, unterschieden werden. Diese Nummer ist eine Hexadezimalzahl, in der Regel 04. Sie muss ein Vielfaches von 4 sein.

Sollten Sie nicht wissen, welchen Wert Sie verwenden müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Ist das ferne Ende dieser Verbindung ein VTAM-Host, entspricht die SAP-Nummer dem Parameter *SAPADDR=* der VTAM-PU-Definition.

Wenn Sie eine Verbindung zu einer AS/400 konfigurieren, ist die SAP-Nummer der Parameter *SSAP* der Leitungsbeschreibung.

### Zusätzliche Verbindungsstationsparameter für X.25 (QLLC):

#### *Ferne X.25-Adresse*

Wenn die Verbindung eine virtuelle Wählverbindung ist, geben Sie die Adresse der fernen DEE als Folge von Hexadezimalziffern ein.

Wenn die Verbindung eine feste virtuelle Verbindung ist, geben Sie die Kanal-ID ein, die die virtuelle Verbindung bezeichnet, die die Verbindungsstation verwenden soll. Für Kanal-IDs werden die Nummern 1 bis maximal 1024 vergeben. Wenn Sie nur über eine feste virtuelle Verbindung verfügen, ist die Kanal-ID dieser Verbindung wahrscheinlich 1.

### Zusätzliche Verbindungsstationsparameter für MPC:

#### *MPC-Gruppe*

Der in der MPC-Treiberkonfiguration angegebene MPC-Gruppenname (MultiPath Channel) zur Bezeichnung eines bestimmten Kanals.

## Verbindungsstationen definieren

### Zusätzliche Verbindungsstationsparameter für Enterprise Extender (HPR/IP):

#### *Name des fernen IP-Hosts*

Ferner Hostname des Zielknotens für diese Verbindung. Für den Hostnamen ist eine der folgenden Angaben möglich. Der Parameter *Protokoll* für den von dieser Verbindung verwendeten Port bestimmt, ob die Adresse das IPV4- oder IPV6-Format hat.

- IPV4-Adresse in Schreibweise mit Trennzeichen (z. B. 193.1.11.100)
- IPV6-Hexadezimaladresse mit Doppelpunkt (z. B. 2001:0db8:0000:0000:0000:0000:1428:57ab oder 2001:db8::1428:57ab)
- Name (z. B. newbox.this.co.uk)
- Aliasname (z. B. newbox)

Wenn Sie einen Namen oder Aliasnamen angeben, muss das Linux-System diesen (mit der lokalen TCP/IP-Konfiguration oder einem Domänen-namensserver) in einen vollständig qualifizierten Namen auflösen können.

### Parameter für unabhängigen LU-Datenverkehr

Wenn Sie die Verbindungsstation für unabhängige LUs (LUs des Typs 6.2 für APPC-, 5250- oder CPI-C-Anwendungen) konfigurieren wollen, benötigen Sie die folgenden Informationen:

#### *Name des fernen Knotens*

Der vollständig qualifizierte CP-Name des fernen Knotens.

Wenn das ferne System ein VTAM-Host ist, entspricht der Netzname (die ersten acht Zeichen des vollständig qualifizierten Namens) dem Parameter *NETID* der VTAM-Startliste. Die letzten acht Zeichen entsprechen dem Parameter *SSCPNAME* der VTAM-Startliste.

**Anmerkung:** Wenn Sie den Namen eines neuen fernen Knotens eingeben, können Sie eine Definition für den fernen Knoten hinzufügen, um Partner-LUs für den neuen fernen Knoten zu definieren. (Wenn der lokale Knoten ein LEN-Knoten ist, müssen Sie die Art des fernen Knotens nicht definieren. Das Feld *Art des fernen Knotens* findet in diesem Fall keine Anwendung.)

Wollen Sie einen fernen Knoten auf diese Weise definieren, geben Sie für die Art des fernen Knotens einen anderen Wert als *Feststellen* und den Namen des fernen Knotens an.

Alternativ dazu können Sie *Feststellen* angeben, müssen dann jedoch das Feld für den Namen des fernen Knotens leer lassen. In diesem Fall kann die Verbindungsstation von jedem Nachbarknoten verwendet werden. Die Option *Feststellen* ist nicht verfügbar, wenn der lokale Knoten ein LEN-Knoten ist.

#### *Art des fernen Knotens*

Die Stufe der APPN-Unterstützung des fernen Knotens, auf den über diese Verbindungsstation zugegriffen wird. (Dieses Feld findet nur Anwendung, wenn der lokale Knoten ein Endknoten oder Netzknoten ist.)

Sollten Sie nicht wissen, ob der ferne Knoten ein LEN-Knoten, Endknoten oder Netzknoten ist, wählen Sie die Option *Feststellen* aus. Die Feststellung der Stufe der APPN-Unterstützung auf dem fernen Knoten kann die Verbindungsaktivierung leicht verzögern. Wenn Ihnen die Art des Knotens

bekannt ist, sollten Sie sie deshalb angeben. Damit tragen Sie gleichzeitig zur Gewährleistung der Konsistenz der Netzkonfiguration bei.

Wenn die Verbindungsstation bei Anforderung aktiviert wird, können Sie die Option Feststellen nicht auswählen.

Ist der lokale Knoten ein LEN-Knoten, findet dieses Feld keine Anwendung.

### *Art der Zweigverbindung*

Die Art der Verbindung zum fernen Knoten, auf den über diese Verbindungsstation zugegriffen wird. (Dieses Feld findet nur Anwendung, wenn der lokale Knoten ein Zweignetzknoten ist.)

Ist der ferne Knoten ein Netzknoten innerhalb des zentralen APPN-Hauptnetzes, wählen Sie Verbindung zu übergeordneter Einheit (zum Zentralnetz) aus. Wenn der ferne Knoten ein Endknoten innerhalb des Zweignetzes ist, wählen Sie Verbindung zu untergeordneter Einheit (innerhalb des Zweignetzes) aus.

Ist der ferne Knoten als Netzknoten konfiguriert, wird die Art der Zweigverbindung automatisch auf Verbindung zu übergeordneter Einheit (zum Zentralnetz) gesetzt und kann nicht geändert werden.

## Parameter für abhängigen LU-Datenverkehr

Diese Parameter finden bei Enterprise-Extender-Verbindungen (HPR/IP) keine Anwendung, weil diese Verbindungsarten nur unabhängigen Datenverkehr unterstützen.

Beim Konfigurieren einer Verbindungsstation für abhängigen LU-Datenverkehr wird automatisch eine geeignete PU erstellt, die denselben Namen wie die Verbindungsstation hat.

Wenn Sie eine Verbindungsstation für abhängige LUs (LUs der Typen 0–3 für 3270- oder LUA-Anwendungen) konfigurieren wollen, benötigen Sie die folgenden Informationen:

### *ID des lokalen Knotens*

Ein Wert, über den der lokale Knoten im SNA-Netz identifiziert werden kann.

In der Regel können Sie für alle Verbindungen eines Knotens dieselbe Knoten-ID (den Standardwert) verwenden. Wenn jedoch mehr als 255 abhängige LUs in der Lage sein sollen, auf einen bestimmten Host zuzugreifen, müssen Sie mehrere Verbindungsstationen zum Host mit jeweils bis zu 255 abhängigen LUs und einer eigenen lokalen Knoten-ID konfigurieren.

Stellen Sie in Abstimmung mit dem zuständigen Planungsbeauftragten für das SNA-Netz sicher, dass der ferne Knoten so konfiguriert wird, dass er die ID des lokalen Knotens erkennen kann.

In einer VTAM-Konfiguration müssen die ersten drei Stellen mit dem Parameter *IDBLK* der PU-Definition und die letzten fünf Stellen mit dem Parameter *IDNUM* übereinstimmen.

Auf einer AS/400 wird die Knoten-ID über den Parameter *EXCHID* konfiguriert.

### *ID des fernen Knotens*

Die Knoten-ID für die ferne Verbindungsstation. (Dieser Parameter ist optional; er ist nur verfügbar, wenn Sie den Zugriff auf diese Verbindungsstation einschränken müssen.) Wenn Sie die ID des fernen Knotens ange-

## Verbindungsstationen definieren

ben, wird die Verbindung nur aktiviert, sofern die Knoten-ID des fernen Knotens dem in dieser Definition festgelegten Wert entspricht. Dies kann nützlich sein, wenn Sie für einen Port mit Wählleitung mehrere Verbindungsstationen konfiguriert haben, denn auf diese Weise können die Verbindungsstationen unterschieden werden, wenn sie von den fernen Knoten aktiviert werden. Verbindungsstationen können auch über den CP-Namen des fernen Knotens unterschieden werden. Für ferne Knoten, die beim Aktivieren einer Verbindung nicht ihren CP-Namen senden, muss stattdessen die ID des fernen Knotens verwendet werden.

Wenn Sie die ID des fernen Knotens nicht angeben, wird die Knoten-ID des fernen Knotens beim Aktivieren der Verbindung nicht überprüft.

### *Funktion des fernen Knotens*

Die Funktion des fernen (benachbarten) Knotens. Hierbei kann es sich um eine der folgenden Funktionen handeln:

**Host** Die Verbindungsstation unterstützt abhängige LUs (z. B. 3270-LUs), die für Sitzungen mit einem Host verwendet werden (dies ist meistens der Fall). Führt die Verbindungsstation zu einem Knoten, der über SNA Gateway oder DLUR Hostkonnektivität bereitstellt, sollte die Funktion des Nachbarknotens auf Host gesetzt werden, auch wenn die Verbindung nicht direkt zu einem Host führt.

### **Untergeordnet (SNA Gateway)**

Die Verbindungsstation führt zu einem untergeordneten Knoten, der über die SNA-Gateway-Funktionen des lokalen Knotens mit einem Host kommuniziert. (Aus Sicht des Hosts befinden sich die LUs des untergeordneten Knotens auf dem lokalen Knoten).

### **Untergeordnet (DLUR)**

Die Verbindungsstation führt zu einem untergeordneten Knoten, der über die DLUR-Funktion des lokalen Knotens mit einem Host kommuniziert. (Aus Sicht des Hosts befinden sich die LUs des untergeordneten Knotens auf dem lokalen Knoten.)

Solche Verbindungen können nur verwendet werden, wenn der lokale Knoten ein APPN-Netz-knoten ist.

### *Name der untergeordneten PU*

Der dem untergeordneten Knoten zugeordnete PU-Name. Dieser Wert muss mit dem auf dem Host für den untergeordneten Knoten konfigurierten PU-Namen übereinstimmen. Sollten Sie nicht wissen, welchen Wert Sie für diesen Namen angeben müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Dieses Feld findet nur Anwendung, wenn Sie angegeben haben, dass die Verbindungsstation zu einer untergeordneten PU führt, die über die DLUR-Funktion des lokalen Knotens mit einem Host kommuniziert. Dies können Sie festlegen, indem Sie für das Feld *Funktion des fernen Knotens* den Wert Untergeordnet (DLUR) angeben.

Weitere Informationen hierzu finden Sie im Abschnitt „DLUR-PU definieren“ auf Seite 77.

### *Name des übergeordneten DLUS*

Der vollständig qualifizierte LU-Name der Host-LU, die DLUS (den LU-Server, auf den die untergeordnete PU zugreifen soll) unterstützt. Sollten Sie nicht wissen, welchen Wert Sie für diesen Namen angeben müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Dieses Feld findet nur Anwendung, wenn Sie angegeben haben, dass die Verbindungsstation zu einer untergeordneten PU führt, die über die DLUR-Funktion des lokalen Knotens mit einem Host kommuniziert. Dies können Sie festlegen, indem Sie für das Feld *Funktion des fernen Knotens* den Wert *Untergeordnet (DLUR)* angeben.

### Zusätzliche Konfiguration

Führen Sie nach dem Konfigurieren der Verbindungsstation die folgenden Konfigurationsaufgaben aus:

- Wie Sie eine DLUR-PU definieren, können Sie im Abschnitt „DLUR-PUs definieren“ nachlesen.
- Wie Sie Durchgriffsdienste definieren, können Sie in Kapitel 8, „Durchgriffsdienste konfigurieren“, auf Seite 113 nachlesen.
- Wie Sie Unterstützung für bestimmte Benutzeranwendungen konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.
- Wie Sie Unterstützung für die APPC-Kommunikation definieren, können Sie in Kapitel 6, „APPC-Kommunikation konfigurieren“, auf Seite 85 nachlesen.

---

### DLUR-PUs definieren

Normalerweise erfordert eine abhängige LU-Sitzung eine direkte DFV-Verbindung zum Host. Wenn in einem APPN-Netz viele Knoten (einschließlich eines Hostknotens) verbunden sind, haben einige von ihnen unter Umständen keine Direktverbindung zum Host, sondern nur eine indirekte Verbindung über einen anderen Knoten. LUs auf diesen indirekt mit dem Host verbundenen Knoten können keine abhängigen LU-Sitzungen mit dem Host aufbauen.

Dependent LU Requester (DLUR) ist ein APPN-Feature, das diese Einschränkung aufhebt.

In diesem Abschnitt wird das Konfigurieren einer DLUR-PU erläutert, die Konnektivität zu einem Host bereitstellt. Durch das Konfigurieren einer DLUR-PU kann der lokale Knoten DLUR-Dienste anbieten.

Die DLUR-Funktion auf einem APPN-Knoten (z. B. einem CS-Linux-Knoten) arbeitet mit DLUS (Dependent LU Server) auf dem Host zusammen, um Sitzungen von abhängigen LUs auf dem DLUR-Knoten über das APPN-Netz zum DLUS-Host weiterzuleiten. Die Route zum Host kann sich über mehrere Knoten erstrecken und von den APPN-Funktionen für Netzverwaltung, für dynamische Ressourcenlokalisierung und Routenberechnung profitieren. DLUR muss auf dem Knoten verfügbar sein, auf dem sich die LUs befinden. DLUS muss auf dem Hostknoten verfügbar sein. Auf den Zwischenknoten der Sitzungsroute ist das Feature DLUR nicht erforderlich.

Wenn der CS-Linux-DLUR-Knoten ein Netzknoten ist, kann er auch Durchgriff-DLUR-Funktionen für abhängige LUs auf untergeordneten Computern, die mit dem CS-Linux-Knoten verbunden sind, bereitstellen. Diese LUs können ebenso wie knoteninterne LUs DLUR auf dem CS-Linux-Knoten nutzen, um über das Netz auf den Host zuzugreifen.

Um Durchgriff-DLUR-Dienste für einen untergeordneten Knoten bereitstellen zu können, müssen Sie zunächst (auf dem lokalen Knoten) den PU-Namen konfigurieren.

## DLUR-PUs definieren

ren, der dem untergeordneten Knoten zugeordnet ist. Dieser Wert muss mit dem auf dem Host für den untergeordneten Knoten konfigurierten PU-Namen übereinstimmen.

Verwenden Sie für das Konfigurieren einer DLUR-PU eine der folgenden Methoden:

### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **Konnektivität** und dann die Option **Neue DLUR-PU** aus.

### Befehlszeilenverwaltungsprogramm

Geben Sie den folgenden Befehl ein:

```
define_internal_pu
```

## Konfigurationsparameter für DLUR-PUs

Für die Konfiguration einer DLUR-PU sind die folgenden Parameter erforderlich:

### *PU-Name*

Geben Sie für jede DLUR-PU auf dem lokalen Knoten einen PU-Namen an. Dieser Name muss mit dem auf dem Host konfigurierten PU-Namen übereinstimmen. (Communications Server für Linux sendet den PU-Namen und die PU-ID an den Host, um die PU zu identifizieren. Der Host identifiziert die PU normalerweise über ihren PU-Namen. Kann er keinen passenden PU-Namen finden, verwendet er die PU-ID für die Identifizierung.)

### *DLUS-Name*

Der vollständig qualifizierte LU-Name der Host-LU, die DLUS unterstützt.

Voraussetzung für die Verwendung von DLUR ist, dass die DLUR-Komponente von Communications Server für Linux eine LU-LU-Sitzung mit DLUS auf dem Host aufbaut.

Den Namen der Host-LU kann Ihnen der zuständige Planungsbeauftragte für das SNA-Netz nennen.

### *Name des Ausweich-DLUS*

Dieser Parameter ist optional. Der vollständig qualifizierte LU-Name einer Ausweichhost-LU, die verwendet werden kann, wenn die mit *DLUS-Name* angegebene LU nicht verfügbar ist.

Den Namen der Host-LU kann Ihnen der zuständige Planungsbeauftragte für das SNA-Netz nennen.

*PU-ID* Die PU-ID der PU auf dem lokalen Knoten, die die Konnektivität zum Host unterstützt. Die PU-ID besteht aus zwei hexadezimalen Zeichenfolgen, einer mit drei Stellen (der so genannten Blocknummer) und einer mit fünf Stellen.

Jede abhängige LU ist einer PU zugeordnet. Sowohl die PU als auch die LU ist auf dem Host konfiguriert. Sie müssen für jede PU auf dem CS-Linux-Knoten eine DLUR-PU definieren. Die PU-ID muss mit der auf dem Host für diese PU konfigurierten PU-ID übereinstimmen.

In vielen Fällen ist die PU-ID mit der Knoten-ID identisch. Die Knoten-ID ist deshalb der Standardwert. Wenn jedoch mehr als 255 abhängige LUs in der Lage sein sollen, auf einen bestimmten Host zuzugreifen, müssen Sie mehrere DLUR-PUs mit jeweils bis zu 255 abhängigen LUs und einer eigenen PU-ID konfigurieren.

Sollten Sie nicht sicher sein, auf welchen Wert Sie dieses Feld setzen müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

In einer VTAM-Konfiguration müssen die ersten drei Stellen mit dem Parameter *IDBLK* der PU-Definition und die letzten fünf Stellen mit der Einstellung für *IDNUM* übereinstimmen.

#### *Aktiv bei Initialisierung*

Information, ob die DLUR-PU beim Starten des Knotens automatisch aktiviert werden soll. Wenn Sie diese Option nicht definieren, muss die DLUR-PU manuell gestartet werden.

#### *Komprimierung unterstützt*

Gibt an, ob für LU-0-3-Sitzungen, die diese PU verwenden, die Datenkomprimierung unterstützt wird. Bei Festlegung dieser Option wird die Komprimierung verwendet, wenn sie vom Host angefordert wird. Wenn Sie diese Option nicht festlegen, wird keine Komprimierung verwendet.

#### *Unbegrenzte Verbindungsversuche zum DLUS*

Gibt an, ob Communications Server für Linux wiederholt versucht, die Verbindung zum DLUS herzustellen, wenn der erste Versuch scheitert. Wenn Sie diese Option festlegen, startet Communications Server für Linux unbegrenzt viele Versuche, nachdem der erste Versuch fehlgeschlagen ist. Wenn Sie diese Option nicht festlegen, wird der Versuch nur einmal wiederholt.

## Parameter für Durchgriff-DLUR für untergeordnete Knoten

Wenn Sie Communications Server für Linux so konfigurieren wollen, dass für den Transport des Verkehrs zwischen abhängigen LUs untergeordneter Knoten und einem Host Durchgriff-DLUR verwendet wird, benötigen Sie die folgenden Informationen:

#### *Name der untergeordneten PU*

Der dem untergeordneten Knoten zugeordnete PU-Name. Der PU-Name muss mit dem auf dem Host konfigurierten PU-Namen übereinstimmen.

Ein untergeordneter Knoten kann mehrere PUs unterstützen. In einem solchen Fall ist jede untergeordnete PU einer anderen Verbindung zugeordnet, so dass Sie zwischen dem CS-Linux-DLUR-Knoten und dem untergeordneten Knoten mehrere Verbindungen konfigurieren und für jede Verbindung den Namen der untergeordneten PU kennen müssen.

Welche PU-Namen den untergeordneten Knoten zugeordnet sind, erfahren Sie beim zuständigen Planungsbeauftragten für das SNA-Netz.

#### *DLUS-Name*

Der vollständig qualifizierte LU-Name der Host-LU, die DLUS unterstützt. Voraussetzung für die Verwendung von DLUR ist, dass die DLUR-Komponente von Communications Server für Linux eine LU-LU-Sitzung mit DLUS auf dem Host aufbaut.

Den LU-Namen für den Host-DLUS erfahren Sie beim zuständigen Planungsbeauftragten für das SNA-Netz.

### Zusätzliche Konfiguration

Führen Sie nach der DLUR-Konfiguration die folgenden Konfigurationsaufgaben aus:

- Wie Sie LUs für DLUR konfigurieren, können Sie im Abschnitt „DLUR konfigurieren“ auf Seite 124 nachlesen.
- Wie Sie weitere Durchgriffsdienste definieren, können Sie in Kapitel 8, „Durchgriffsdienste konfigurieren“, auf Seite 113 nachlesen.
- Wie Sie Unterstützung für bestimmte Benutzeranwendungen konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.
- Wie Sie Unterstützung für die APPC-Kommunikation definieren, können Sie in Kapitel 6, „APPC-Kommunikation konfigurieren“, auf Seite 85 nachlesen.

---

## Kapitel 5. Abhängige LUs konfigurieren

Dieses Kapitel enthält Anweisungen für das Konfigurieren von LUs und LU-Pools zur Unterstützung von Benutzeranwendungen, die mit der 3270-, TN3270- und LUA-Kommunikation arbeiten. Für diese Anwendungen müssen Sie abhängige LUs konfigurieren.

Bevor Sie die in diesem Kapitel beschriebenen Ressourcen konfigurieren können, müssen Sie die folgenden Konfigurationsschritte ausführen:

- Konfigurieren Sie den Knoten wie im Abschnitt „Knoten konfigurieren“ auf Seite 58 beschrieben.
- Konfigurieren Sie die Konnektivität wie in Kapitel 4, „Konnektivitätskomponenten definieren“, auf Seite 63 beschrieben. Für 3270, TN3270 und LUA müssen Sie die Verbindung für Unterstützung von abhängigem LU-Datenverkehr konfigurieren.

Wenn Sie übergeordnetes SNA Gateway oder DLUR verwenden, müssen Sie keine Direktverbindung zum Host konfigurieren. Weitere Informationen hierzu finden Sie im Abschnitt „SNA Gateway konfigurieren“ auf Seite 122 und im Abschnitt „DLUR-PU definieren“ auf Seite 77.

---

### LUs der Typen 0–3 definieren

Zur Unterstützung der Kommunikation mit einem Hostsystem müssen Sie abhängige LUs der Typen 0–3 konfigurieren. Anhand der Informationen in diesem Abschnitt können Sie eine LU für die Unterstützung der 3270- oder LUA-Kommunikation definieren. Wenn Sie in einem Schritt mehrere LUs konfigurieren wollen, können Sie auch einen Bereich von LUs definieren.

Verwenden Sie zum Konfigurieren von LUs der Typen 0–3 eine der folgenden Methoden:

#### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters einen der folgenden Einträge aus.

- **3270** und die Option **Neue 3270-Anzeige-LU** oder **Neue 3270-Drucker-LU**
- **LUA** und die Option **Neue LUA-LU**
- **TN Server** und die Option **Neue Host-LU**

#### Befehlszeilenverwaltungsprogramm

Setzen Sie einen der folgenden Befehle ab:

```
define_lu_0_to_3
```

```
define_lu_0_to_3_range
```

Im erweiterten Dialog können Sie den Zugriff auf einen bestimmten SSCP beschränken oder ein Zeitlimit für Inaktivität angeben.

### Konfigurationsparameter für LUs der Typen 0–3

Für die Konfiguration von LUs der Typen 0–3 sind die folgenden Parameter erforderlich:

#### *LU-Name*

Ein aus 1–8 Zeichen bestehender LU-Name (für eine einzelne LU) oder ein aus 1–5 Zeichen bestehender Basisname. (Für einen LU-Bereich wird zum Ableiten der Namen aller definierten LUs zum Basisnamen ein Präfix hinzugefügt.)

Der LU-Name wird nur lokal verwendet. Er muss nicht mit einem auf dem Host definierten Namen übereinstimmen.

#### *Hostverbindungsstation/DLUR-PU*

Die Verbindungsstation, über die die Verbindung zum Host hergestellt wird. Die LU-Definition gehört zu der von Ihnen ausgewählten Verbindungsstation. (Befindet sich die abhängige LU auf einem Knoten, der DLUR unterstützt, gibt dieses Feld die DLUR-PU an, die Konnektivität zum Host bereitstellt.)

#### *LU-Nummern*

Eine LU-Nummer oder ein Bereich von LU-Nummern. LU-Nummern können im Bereich 1–255 liegen.

Die Nummern müssen denen der Host-VTAM-Konfiguration entsprechen. Sollten Sie nicht wissen, welche Nummern auf dem Host konfiguriert sind, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

#### *LU-Typ*

Hierbei kann es sich um einen der folgenden LU-Typen handeln:

- Wenn Sie diese LU mit TN Server und DDDLU verwenden, ist die LU möglicherweise nicht auf dem Host konfiguriert. In diesem Fall wird der hier von Ihnen angegebene LU-Typ verwendet, um die LU auf dem Host dynamisch zu definieren. Geben Sie Unbeschränkt (bzw. bei der Befehlszeilenkonfiguration unknown) an, wenn der LU-Modelltyp in Übereinstimmung mit dem vom untergeordneten TN3270-Client geforderten Typ definiert werden soll.
- Geben Sie für eine 3270-Anzeige-LU das Modell mit der entsprechenden Anzeigegröße an:
  - 3270 Modell 2 (80x24)
  - 3270 Modell 3 (80x32)
  - 3270 Modell 4 (80x43)
  - 3270 Modell 5 (132x27)
- Geben Sie für eine Drucker-LU einen der folgenden Werte an:
  - 3270-Drucker
  - SCS-Drucker
- Wenn Sie den LU-Typ nicht kennen oder wenn die LU zur Unterstützung eines SNA-Gateways vom lokalen Knoten zum Host (einer übergeordneten LU) verwendet wird bzw. für eine LUA-Anwendung bestimmt ist, geben Sie Unbeschränkt an (im Befehlszeilenverwaltungsprogramm unknown).

Der LU-Typ sollte mit der Konfiguration der LU auf dem Host übereinstimmen. Ggf. hat der auf dem Host konfigurierte LU-Typ Vorrang.

Communications Server für Linux sendet abhängig von dem Wert, den Sie angeben, im DDDLUNMVT eine der folgenden Zeichenfolgen an den Host, die zu Werten in den Standard-VTAM-Tabellen passen:

3270002 für 3270, Modell 2  
3270003 für 3270, Modell 3  
3270004 für 3270, Modell 4  
3270005 für 3270, Modell 5  
3270DSC für 3270-Drucker  
3270SCS für SCS-Drucker  
3270000 für RJE-Workstation  
327000n für Unbeschränkt mit einem TN3270-Client (*n* ist Modellnummer (2–5), die der Client angibt)  
327000@ für Unbeschränkt mit einem LUA-Client

### *LU im Pool*

Information, ob die LU einem LU-Pool zugeordnet ist.

**Anmerkung:** Wenn Sie diese LU einem LU-Pool zuordnen und dieser LU eine Benutzersitzung zuordnen, verwendet die Benutzersitzung diese LU, sofern sie verfügbar ist. Andernfalls verwendet die Sitzung eine beliebige freie LU aus dem Pool, als hätten Sie die Sitzung dem LU-Pool und nicht dieser speziellen LU zugeordnet. Falls der Benutzer nur eine angegebene LU verwenden soll, so dass die Benutzersitzung nicht aufgebaut wird, wenn die LU bereits benutzt wird, müssen Sie sicherstellen, dass die LU nicht in einem Pool enthalten ist.

### *Poolname*

Der Name des LU-Pools.

## Zusätzliche Konfiguration

Führen Sie nach dem Konfigurieren der LU der Typen 0–3 die folgenden Konfigurationsaufgaben aus:

- Wenn Sie einen Pool abhängiger LUs für ein 3270-Terminal, für TN3270 oder für LUA verwenden wollen, definieren Sie den Pool wie im Abschnitt „LU-Pools definieren“ beschrieben.
- Definieren Sie für TN3270 Zugriffseinträge für TN3270-Clients. Gehen Sie dabei wie im Abschnitt „TN Server konfigurieren“ auf Seite 113 beschrieben vor.

---

## LU-Pools definieren

Für 3270, TN3270 und LUA können Sie LU-Pools definieren, um die Benutzerkonfiguration zu vereinfachen und eine größere Flexibilität beim Aufbauen von Hostsitzungen zu erzielen.

**Anmerkung:** Sie können einer bestimmten LU oder einem LU-Pool eine Benutzersitzung zuordnen.

- Wenn Sie die Benutzersitzung einer bestimmten LU zuordnen, die in einem Pool enthalten ist, verwendet die Sitzung diese LU, sofern sie verfügbar ist. Andernfalls verwendet die Sitzung eine beliebige freie LU aus dem Pool, als hätten Sie die Sitzung dem LU-Pool und nicht dieser speziellen LU zugeordnet.

## LU-Pools definieren

- Falls der Benutzer nur eine angegebene LU verwenden soll, so dass die Benutzersitzung nicht aufgebaut wird, wenn die LU bereits benutzt wird, müssen Sie sicherstellen, dass die LU nicht in einem Pool enthalten ist.

LU-Pools können sich sogar über mehrere CS-Linux-Server erstrecken. Sie müssen auf den verschiedenen Servern lediglich LU-Pools mit identischen Namen definieren. Wenn ein Server ausfällt oder heruntergefahren wird, können Clients, die den LU-Pool verwenden, einen anderen Server nutzen. Die Verwendung von LU-Pools vereinfacht außerdem die Clientkonfiguration und Kapazitätssteigerungen durch das Hinzufügen eines weiteren Servers oder von LUs auf einem vorhandenen Server.

Im Fenster für LU-Pools können Sie alle LU-Pools für die CS-Linux-Domäne anzeigen. Dieses Fenster listet die auf dem System konfigurierten LU-Pools auf. Sie können in diesem Fenster LUs auswählen und zu einem LU-Pool hinzufügen. Die einzelnen in einem LU-Pool enthaltenen LUs werden unterhalb des LU-Pools angezeigt.

Eine LU ist wie folgt gekennzeichnet:

- 3270-Anzeige-LU
- Unbeschränkte LU
- SCS-Drucker
- 3270-Drucker

Ein Pool darf nicht LUs verschiedener Typen enthalten. (Sie können beispielsweise nicht Anzeige- und Drucker-LUs zum selben Pool hinzufügen.) Es ist unwahrscheinlich, dass Sie einen Pool mit Drucker-LUs benötigen werden, es sei denn, Sie unterstützen TN3270E-Clients.

Verwenden Sie für das Konfigurieren eines LU-Pools eine der folgenden Methoden:

### **Motif-Verwaltungsprogramm**

Wählen Sie im Menü **Fenster** des Knotenfensters den Eintrag **LU-Pools** und dann die Option **Neu** aus, um einen Pool hinzuzufügen.

### **Befehlszeilenverwaltungsprogramm**

Geben Sie den folgenden Befehl ein:

```
define_lu_pool
```

## **Konfigurationsparameter für LU-Pools**

Für die Konfiguration eines LU-Pools sind die folgenden Parameter erforderlich:

*Name* Ein Name, über den der LU-Pool identifiziert werden kann. Dieses Feld findet nur Anwendung, wenn Sie einen neuen LU-Pool hinzufügen. Den Namen eines vorhandenen Pools können Sie nicht ändern.

### *Zugeordnete LUs*

LUs, die zum Pool hinzugefügt werden sollen. Eine LU kann immer nur zu jeweils einem Pool gehören.

---

## Kapitel 6. APPC-Kommunikation konfigurieren

APPC-Anwendungen, 5250-Emulationsprogramme und CPI-C-Anwendungen erfordern, dass Sie zunächst APPC konfigurieren. Eine APPC-Anwendung benutzt die LU-6.2-Ressourcen des Knotens, um in einem bestimmten Modus mit anderen APPC- oder CPI-C-Anwendungen auf einem Host oder einem Peercomputer zu kommunizieren.

Wenn die Anwendungen mit CPI-C arbeiten, müssen Sie unter Umständen nach der APPC-Konfiguration weitere Konfigurationsschritte für CPI-C ausführen. Eine CPI-C-Anwendung benutzt die LU-6.2-Ressourcen und die Modusressourcen des Knotens, um mit einer anderen APPC- oder CPI-C-Anwendung auf einem Host oder einem Peercomputer zu kommunizieren. Für eine CPI-C-Anwendung müssen Sie die gleichen Ressourcen wie für eine APPC-Anwendung definieren. Wenn das Transaktionsprogramm auf dem CS-Linux-Computer das aufrufende Transaktionsprogramm (das den Dialog startende TP) ist, müssen Sie für dieses Transaktionsprogramm möglicherweise einen oder mehrere Einträge mit Nebeninformationen definieren. Gehen Sie dazu wie im Abschnitt „CPI-C-Nebeninformationen definieren“ auf Seite 105 beschrieben vor. Jeder dieser Einträge stellt Informationen zu einem Partnertransaktionsprogramm, die für den Zugriff auf dieses Programm verwendeten LU- und Modusressourcen sowie alle benötigten Sicherheitsinformationen bereit.

Die erforderlichen Konfigurationsschritte für APPC richten sich danach, ob der LU-6.2-Verkehr abhängig oder unabhängig ist. Sofern der ferne Knoten kein Host ist, müssen Sie unabhängigen Verkehr verwenden. Wenn der ferne Knoten ein Host ist, können Sie abhängigen oder unabhängigen Verkehr verwenden.

Bevor Sie die APPC-Kommunikation konfigurieren können, müssen Sie die folgenden Konfigurationsschritte ausführen:

- Konfigurieren Sie den Knoten wie im Abschnitt „Knoten konfigurieren“ auf Seite 58 beschrieben.
- Konfigurieren Sie die Konnektivität wie in Kapitel 4, „Konnektivitätskomponenten definieren“, auf Seite 63 beschrieben.

**Anmerkung:** In einem APPN-Netz ist es möglich, über nur eine Verbindungsstation zu einem benachbarten Netzknoten mit allen fernen Knoten im Netz zu kommunizieren. Sie müssen deshalb nicht zu jedem fernen Knoten eine gesonderte Verbindungsstation konfigurieren.

In vielen Fällen können APPC-Anwendungen die Steuerpunkt-LU des lokalen und des fernen Knotens und einen Standardmodus verwenden. Sollte dies für Sie zutreffen, ist die APPC-Konfiguration beendet. Weitere Konfigurationsschritte sind nicht erforderlich.

Mit Hilfe der folgenden Schritte können Sie die APPC-Kommunikation auf dem lokalen Knoten konfigurieren. Je nach Art des lokalen und fernen Knotens und der Anwendung ist es möglich, dass Sie diese Schritte nicht ausführen müssen.

1. Definieren Sie eine lokale LU wie im Abschnitt „Lokale LUs definieren“ auf Seite 86 beschrieben.
2. Definieren Sie einen fernen Knoten wie im Abschnitt „Ferne Knoten definieren“ auf Seite 88 beschrieben.

## APPC-Kommunikation konfigurieren

3. Definieren Sie eine Partner-LU wie im Abschnitt „Partner-LUs definieren“ auf Seite 89 beschrieben.
4. Definieren Sie ein aufrufbares Transaktionsprogramm wie im Abschnitt „TPs definieren“ auf Seite 93 beschrieben.
5. Definieren Sie einen Modus wie im Abschnitt „Modi und Serviceklassen definieren“ auf Seite 100 beschrieben.
6. Definieren Sie CPI-C-Nebeninformationen wie im Abschnitt „CPI-C-Nebeninformationen definieren“ auf Seite 105 beschrieben.
7. Definieren Sie APPC-Sicherheit wie im Abschnitt „APPC-Sicherheit konfigurieren“ auf Seite 108 beschrieben.
8. Wie Sie die 5250-Kommunikation konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.

---

## Lokale LUs definieren

In vielen Fällen können Anwendungen die Steuerpunkt-LU des lokalen Knotens benutzen, die beim Konfigurieren des Knotens automatisch definiert wird. Dies ist die Standard-LU. Wenn eine Anwendung keine bestimmte LU angibt, kann sie diese LU verwenden. Benutzt die Anwendung die Standard-LU, müssen Sie keine lokale LU definieren. Diesbezügliche Informationen finden Sie in der Dokumentation zur APPC-Anwendung. Entsprechende Hinweise kann auch der Anwendungsprogrammierer geben.

Wenn Sie abhängige LUs des Typs 6.2 für APPC- oder CPI-C-Anwendungen konfigurieren, können Sie sie als LUs des Standardpools definieren. Einer Anwendung, die keine bestimmte lokale LU angibt, wird eine nicht benutzte LU des Standard-LU-Pools zugeordnet.

Abhängige LUs 6.2 können Sie als Standard-LUs definieren. (Sie können Standard-LUs auf mehr als einem Knoten definieren.) Eine Anwendung, die eine Standard-LU anfordert, kann jeder verfügbaren dieser LUs zugeordnet werden. Die LUs müssen sich nicht auf demselben Computer befinden wie die Anwendung. Wenn Sie für die Anwendungen jedoch Partner-LUs definieren, müssen Sie diese auf allen Knoten definieren, auf denen Standard-LUs definiert sind, damit die Anwendung über eine der auf einem beliebigen Knoten definierten lokalen Standard-LUs Verbindung zur richtigen Partner-LU aufnehmen kann.

Unabhängige LUs werden von unabhängigem APPC und von 5250 verwendet. An jeder LU-LU-Sitzung nimmt eine lokale LU und eine Partner-LU teil. Als lokale LU können Sie die bereits definierte, dem Knotensteuerpunkt zugeordnete Standard-LU verwenden. Sie können aber auch neue lokale LUs konfigurieren. Die Partner-LU muss nicht konfiguriert werden, wenn der CS-Linux-Knoten ein Endknoten oder Netzknoten in einem APPN-Netz ist, da APPN Partner-LUs dynamisch lokalisieren kann. Ist das Netz kein APPN-Netz oder der Knoten ein LEN-Knoten, müssen Sie die Partner-LU allerdings konfigurieren. In diesem Fall müssen Sie den fernen Knoten, auf dem sich die Partner-LU befindet, konfigurieren und dann die Partner-LU auf dem fernen Knoten definieren. (Wenn die Partner-LU die Standard-LU auf dem fernen Knoten ist, müssen Sie sie nicht explizit definieren, denn sie wird beim Definieren des fernen Knotens automatisch hinzugefügt.)

Verwenden Sie für das Konfigurieren einer lokalen APPC-LU eine der folgenden Methoden:

### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Option **Neue unabhängige lokale LU** oder **Neue abhängige lokale LU** aus.

### Befehlszeilenverwaltungsprogramm

Geben Sie den folgenden Befehl ein:

```
define_local_lu
```

In der erweiterten Dialoganzeige können Sie Unterstützung für den Synchronisationspunkt, Kenndaten für das Attach-Routing, Informationen zur Verwendung der Kennworts substitution, Einschränkungen für den Zugriff auf den SSCP, den der LU zugeordneten Systemnamen und Sicherheitsdaten angeben.

## Konfigurationsparameter für lokale LUs

Für die Konfiguration einer lokalen LU sind die folgenden Parameter erforderlich:

#### *LU-Name*

Der LU-Name der lokalen LU.

Sollten Sie nicht wissen, welchen Namen Sie angeben müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Dieser LU-Name ist der zweite Teil des vollständig qualifizierten LU-Namens der lokalen LU. Der erste Teil des vollständig qualifizierten LU-Namens (der Netzname) stimmt immer mit dem ersten Teil des CP-Namens des lokalen Knotens überein.

#### *LU-Aliasname*

Der LU-Aliasname der LU. Wenn Sie keinen Aliasnamen eingeben, wird der LU-Name als Aliasname verwendet.

#### *Hostverbindungsstation/DLUR-PU*

Der Name der Hostverbindungsstation oder der DLUR-PU, zu der die LU gehört. (Dieses Feld findet nur Anwendung, wenn die LU eine abhängige LU ist.)

#### *LU-Nummer*

Die LU-Nummer der abhängigen LU. (Dieses Feld findet nur Anwendung, wenn die LU eine abhängige LU ist.)

#### *Teil des Standardpools*

Information, ob die LU in den Standard-LU-Pool mit abhängigen APPC-LUs aufgenommen werden soll. Einer Anwendung, die keine bestimmte lokale LU angibt, wird eine verfügbare LU des Standard-LU-Pools zugeordnet.

Dieses Feld findet nur Anwendung, wenn die LU eine abhängige LU ist.

## Zusätzliche Konfiguration

Führen Sie nach dem Konfigurieren der lokalen LU die folgenden Konfigurationsschritte aus:

- Wie Sie einen fernen Knoten definieren, können Sie im Abschnitt „Ferne Knoten definieren“ auf Seite 88 nachlesen.
- Wie Sie eine Partner-LU definieren, können Sie im Abschnitt „Partner-LUs definieren“ auf Seite 89 nachlesen.
- Wie Sie ein aufrufbares Transaktionsprogramm definieren, können Sie im Abschnitt „TPs definieren“ auf Seite 93 nachlesen.

## Lokale LUs definieren

- Wie Sie einen Modus definieren, können Sie im Abschnitt „Modi und Serviceklassen definieren“ auf Seite 100 nachlesen.
- Wie Sie CPI-C-Nebeninformationen definieren, können Sie im Abschnitt „CPI-C-Nebeninformationen definieren“ auf Seite 105 nachlesen.
- Wie Sie APPC-Sicherheit definieren, können Sie im Abschnitt „APPC-Sicherheit konfigurieren“ auf Seite 108 nachlesen.
- Wie Sie die 5250-Kommunikation konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.

---

## Ferne Knoten definieren

In folgenden Fällen müssen Sie einen fernen Knoten (und die Partner-LUs auf dem Knoten) definieren:

- Wenn der lokale Knoten ein LEN-Knoten ist, müssen Sie alle fernen Knoten und alle Partner-LUs auf dem fernen Knoten, mit denen der lokale Knoten über APPC kommuniziert, definieren. Ein LEN-Knoten kann Partner-LUs nicht dynamisch lokalisieren. Dies ist nur über die Definition des fernen Knotens möglich.
- Wenn der ferne Knoten ein LEN-Knoten und der lokale Knoten ein Netzknoten ist, der für den LEN-Knoten die Funktion eines Netzknotenservers hat, müssen Sie den LEN-Knoten (mit den Partner-LUs) auf dem Netzknotenserver als fernen Knoten definieren. Durch diese Definition können die Knoten im übrigen APPN-Netz LUs auf dem LEN-Knoten lokalisieren.
- Wenn sich der ferne Knoten in einem anderen APPN-Netz befindet, müssen Sie den fernen Knoten definieren, da dieser nicht dynamisch lokalisiert werden kann.

Wenn Sie den fernen Knoten definieren müssen und dies nicht beim Definieren der Verbindungsstation getan haben, müssen Sie dies nachholen, bevor Sie die APPC-Kommunikation über die Verbindung nutzen können.

Wenn Sie eine Definition für einen fernen Knoten hinzufügen, wird automatisch eine Partner-LU mit demselben Namen hinzugefügt. Dies ist die Steuerpunkt-LU des fernen Knotens. Falls Ihre Anwendung diese Partner-LU benutzt, müssen Sie keine andere Partner-LU hinzufügen. Bei Bedarf können Sie jedoch einen LU-Aliasnamen für die Partner-LU hinzufügen. Klicken Sie zum Hinzufügen eines Aliasnamens doppelt auf die Partner-LU, und geben Sie im Konfigurationsdialog für Partner-LUs den Aliasnamen ein.

Falls sowohl der lokale Knoten als auch der ferne Knoten ein Endknoten oder Netzknoten ist und zu einem APPN-Netz gehört, werden die Partner-LUs bei Bedarf dynamisch lokalisiert. Definieren Sie in diesem Fall nicht den fernen Knoten, auf dem sich die LUs befinden, denn dies kann dazu führen, dass die APPN-Protokolle für die dynamische Lokalisierung von LUs nicht richtig funktionieren.

Um dies zu verhindern, erlaubt die Software Communications Server für Linux keine Definition eines fernen Knotens, wenn zwischen dem Knoten und der Software aktive CP-CP-Sitzungen bestehen (oder in der Vergangenheit bestanden haben). Sollte Communications Server für Linux eine CP-CP-Sitzung mit einem fernen Knoten aufbauen, den Sie bereits früher definiert haben, wird der Eintrag temporär in einen dynamischen Eintrag umgewandelt. Sie sollten den Fehler korrigieren, indem Sie die Definition des fernen Knotens löschen, wenn der Knoten inaktiv ist.

Verwenden Sie für das Konfigurieren eines fernen Knotens eine der folgenden Methoden:

### **Motif-Verwaltungsprogramm**

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Option **Neuer ferner Knoten** aus.

### **Befehlszeilenverwaltungsprogramm**

Setzen Sie zum Definieren eines fernen Knotens den folgenden Befehl ab:

```
define_directory_entry
```

Setzen Sie zum Definieren einer Partner-LU den folgenden Befehl ab:

```
define_partner_lu
```

## **Konfigurationsparameter für ferne Knoten**

Für die Konfiguration eines fernen Knotens ist der folgende Parameter erforderlich:

### *SNA-Netzname des Knotens*

Der vollständig qualifizierte CP-Name des fernen Knotens. Der in dieser Dialoganzeige eingegebene Wert muss mit dem auf dem fernen Knoten konfigurierten CP-Namen übereinstimmen.

## **Zusätzliche Konfiguration**

Führen Sie nach dem Konfigurieren des fernen Knotens die folgenden Konfigurationaufgaben aus:

- Wie Sie eine Partner-LU definieren, können Sie im Abschnitt „Partner-LUs definieren“ nachlesen.
- Wie Sie ein aufrufbares Transaktionsprogramm definieren, können Sie im Abschnitt „TPs definieren“ auf Seite 93 nachlesen.
- Wie Sie einen Modus definieren, können Sie im Abschnitt „Modi und Serviceklassen definieren“ auf Seite 100 nachlesen.
- Wie Sie CPI-C-Nebeninformationen definieren, können Sie im Abschnitt „CPI-C-Nebeninformationen definieren“ auf Seite 105 nachlesen.
- Wie Sie APPC-Sicherheit definieren, können Sie im Abschnitt „APPC-Sicherheit konfigurieren“ auf Seite 108 nachlesen.
- Wie Sie die 5250-Kommunikation konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.

---

## **Partner-LUs definieren**

Wenn der lokale und der ferne Knoten Netzknoten sind oder einer von ihnen ein Netzknoten und der andere ein Endknoten ist und Ihre Anwendung mit einem LU-Namen auf die Partner-LU verweist, müssen Sie die Partner-LU nicht definieren, da sie mit APPN dynamisch lokalisiert werden kann. Wenn Ihre Anwendung jedoch mit einem LU-Aliasnamen auf die zugehörige Partner-LU verweist, sollten Sie für die LU die Definition eines Aliasnamens hinzufügen.

Ist der lokale oder der ferne Knoten ein LEN-Knoten, müssen Sie die Partner-LU als Kind des fernen Knotens definieren, da der LEN-Knoten nicht in der Lage ist, LUs dynamisch zu lokalisieren. Benutzt Ihre Anwendung die Steuerpunkt-LU des fernen Knotens als Partner-LU, müssen Sie keine Partner-LU definieren, denn die Steuerpunkt-LU wurde bereits beim Konfigurieren des fernen Knotens automatisch definiert.

## Partner-LUs definieren

Sie können Platzhalter verwenden, um mehrere Partner-LUs zu definieren, die sich alle auf demselben fernen Knoten befinden und deren Namen mit denselben Zeichen beginnen. Bei Verwendung von Platzhaltern müssen Sie nicht jede Partner-LU einzeln konfigurieren.

Verwenden Sie für das Konfigurieren einer Partner-LU eine der folgenden Methoden:

### Motif-Verwaltungsprogramm

Mit dem Motif-Verwaltungsprogramm können Sie einen Aliasnamen für eine Partner-LU hinzufügen, eine Definition für eine Partner-LU auf einem bestimmten fernen Knoten hinzufügen oder mehrere Partner-LUs unter Verwendung von Platzhaltern definieren. Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Option **Neue Partner-LUs** sowie eine der folgenden Optionen aus.

- **Aliasname der Partner-LU**
- **Partner-LU auf fernem Knoten**
- **Generische Partner-LU auf fernem Knoten**

### Befehlszeilenverwaltungsprogramm

Setzen Sie zum Definieren einer Partner-LU den folgenden Befehl ab:

```
define_partner_lu
```

Setzen Sie zum Definieren eines LEN-Knotens als Partner-LU die folgenden Befehle ab:

```
define_adjacent_len_node
```

```
define_directory_entry
```

## Konfigurationsparameter für Partner-LUs

Für die Konfiguration einer Partner-LU sind die folgenden Parameter erforderlich:

### *Name der Partner-LU*

Der vollständig qualifizierte LU-Name der Partner-LU. Dieser Name muss mit dem auf dem fernen Knoten für diese LU konfigurierten Namen übereinstimmen. Sollten Sie nicht wissen, wie dieser Name lautet, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Dieses Feld findet Anwendung, wenn Sie eine Partner-LU auf einem bestimmten fernen Knoten oder den Aliasnamen einer Partner-LU definieren.

### **Generischer Name der Partner-LU**

Ein Name, der mit den vollständig qualifizierten LU-Namen mehrerer Partner-LUs übereinstimmt. (Dieses Feld findet nur Anwendung, wenn Sie Partner-LUs mit Platzhaltern definieren.) Der generische Name der Partner-LU besteht aus zwei Zeichenfolgen mit einer Länge von jeweils 1–8 Zeichen.

- Die erste Zeichenfolge kann ein vollständiger SNA-Netzname sein, der mit dem ersten Teil des vollständig qualifizierten Namens der Partner-LUs identisch ist, oder ein generisches Präfix, das mit dem Anfang des Netznamens für die Partner-LUs übereinstimmt. Wenn Sie für die erste Zeichenfolge ein generisches Präfix angeben, dürfen Sie im Feld für die zweite Zeichenfolge keinen Wert eingeben. Der generische Eintrag **A**

würde beispielsweise mit allen LUs in den SNA-Netzen übereinstimmen, deren Name A, ANT oder APPN (nicht jedoch BUFFALO oder ZEBRA) lautet.

- Haben Sie für die erste Zeichenfolge einen vollständigen SNA-Netznamen vorgegeben, können Sie auch für die zweite Zeichenfolge einen Wert angeben. (Voraussetzung für die Angabe der zweiten Zeichenfolge ist die Angabe eines gültigen SNA-Netznamens für die erste Zeichenfolge.) Die zweite Zeichenfolge wird als generisches Präfix interpretiert, das mit dem Anfang des zweiten Teils des vollständig qualifizierten Namens der Partner-LUs übereinstimmen muss. Der generische Eintrag **A.F** würde beispielsweise mit den Partner-LU-Namen A.FRED oder A.FREDDY (nicht jedoch mit APPN.FRED oder A.B) übereinstimmen.

Wenn Sie für beide Zeichenfolgen keinen Wert angeben, stimmt die generische Partner-LU-Definition mit allen Namen von Partner-LUs überein.

### *Aliasname*

Ein lokal anzeigbarer Aliasname für die Partner-LU. Sie müssen keinen LU-Aliasnamen angeben, wenn keine lokale Anwendung mit einem LU-Aliasnamen auf die Partner-LU verweist.

Dieses Feld findet Anwendung, wenn Sie eine Partner-LU auf einem bestimmten fernen Knoten oder den Aliasnamen einer Partner-LU definieren.

### *Nicht interpretierter Name*

Der nicht interpretierte Name, den abhängige lokale LUs verwenden, wenn sie den Host auffordern, eine LU-LU-Sitzung zwischen der Partner-LU und der lokalen LU zu starten. Bei Verwendung dieses Namens kann der lokal definierte (und von Anwendungen benutzte) Name der Partner-LU von dem auf dem Host konfigurierten Namen der Partner-LU abweichen.

Die Standardeinstellung für den nicht interpretierten Namen ist der zweite Teil des Namens der Partner-LU. Dies ist in den meisten Fällen korrekt. Sollten Sie nicht sicher sein, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Dieses Feld findet Anwendung, wenn Sie eine Partner-LU auf einem bestimmten fernen Knoten oder den Aliasnamen einer Partner-LU definieren.

### *Unterstützt Parallelsitzungen*

Information, ob die Partner-LU mehr als eine Sitzung zur Zeit unterstützen kann. In den meisten Fällen unterstützt die Partner-LU viele gleichzeitige Sitzungen. Einige LEN-Knoten bieten jedoch keine Unterstützung für Parallelsitzungen.

Dieses Feld findet Anwendung, wenn Sie eine Partner-LU auf einem bestimmten fernen Knoten oder den Aliasnamen einer Partner-LU definieren.

### *Position*

Der vollständig qualifizierte CP-Name des Knotens, auf dem sich die Partner-LU befindet, oder des Knotens, der den Zugriff auf die Partner-LU ermöglicht. Wenn Sie den Namen eines noch nicht definierten fernen Knotens angeben, müssen Sie den Knoten definieren, sofern er nicht dynamisch lokalisiert werden kann.

Dieses Feld findet nur Anwendung, wenn Sie eine Partner-LU auf einem bestimmten fernen Knoten definieren.

### Routenwahl für Verbindungsstationen für eine Partner-LU definieren

Mit Hilfe der Routenwahl für Verbindungsstationen können Sie die Position einer Partner-LU über die Verbindungsstation bestimmen, über die die Partner-LU erreichbar ist.

#### Anmerkung:

1. In einem APPN-Netz können Ressourcen dynamisch lokalisiert werden, so dass kein Routing von Verbindungsstationen erforderlich ist. Sie sollten in einem APPN-Netz kein Routing für Verbindungsstationen verwenden, weil dies die normalen APPN-Routing-Mechanismen umgeht.
2. Für eine Enterprise-Extender-Verbindungsstation (HPR/IP) können Sie das Routing von Verbindungsstationen nicht anwenden. Dies liegt daran, dass bei dieser Verbindungsart der gesamte Datenverkehr über eine RTP-Verbindung transportiert werden muss, die nicht an eine bestimmte Verbindungsstation gebunden ist und auf einen anderen Pfad umschalten kann.

Verwenden Sie zum Konfigurieren der Routenwahl für Verbindungsstationen für eine Partner-LU eine der folgenden Methoden:

#### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Optionen **Neue Partner-LUs** und **Partner-LU auf Verbindungsstation** aus.

#### Befehlszeilenverwaltungsprogramm

Geben Sie den folgenden Befehl ein:

```
define_ls_routing
```

### Parameter für Routenwahl für Verbindungsstationen

Für die Konfiguration der Routenwahl für Verbindungsstationen sind die folgenden Parameter erforderlich:

#### *LU-Name*

Der Name der lokalen LU, die die Verbindungsstation steuert (wenn die Partner-LU über eine bestimmte Verbindungsstation lokalisiert werden soll).

#### *Name der Verbindungsstation*

Der Name der Verbindungsstation.

#### *Name der Partner-LU*

Entweder der vollständig qualifizierte LU-Name der Partner-LU oder ein generischer Name.

- Ein vollständig qualifizierter LU-Name besteht aus zwei Zeichenfolgen mit jeweils 1–8 Zeichen.

Dieser Name muss mit dem auf dem fernen Knoten für diese LU konfigurierten Namen übereinstimmen. Sollten Sie nicht wissen, wie dieser Name lautet, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

- Der generische Name einer Partner-LU stimmt mit den vollständig qualifizierten LU-Namen mehrerer Partner-LUs überein. Der generische Name der Partner-LU besteht aus zwei Zeichenfolgen mit einer Länge von jeweils 1–8 Zeichen.
  - Die erste Zeichenfolge kann ein vollständiger SNA-Netzname sein, der mit dem ersten Teil des vollständig qualifizierten Namens der Partner-LUs identisch ist, oder ein generisches Präfix, das mit dem Anfang des Netznamens für die Partner-LUs übereinstimmt. Wenn Sie für die erste Zeichenfolge ein generisches Präfix angeben, dürfen Sie im Feld für die zweite Zeichenfolge keinen Wert eingeben. Der generische Eintrag **A** würde beispielsweise mit allen LUs in den SNA-Netzen übereinstimmen, deren Name **A**, **ANT** oder **APPN** (nicht jedoch **BUFFALO** oder **ZEBRA**) lautet.
  - Haben Sie für die erste Zeichenfolge einen vollständigen SNA-Netznamen vorgegeben, können Sie auch für die zweite Zeichenfolge einen Wert angeben. (Voraussetzung für die Angabe der zweiten Zeichenfolge ist die Angabe eines gültigen SNA-Netznamens für die erste Zeichenfolge.) Die zweite Zeichenfolge wird als generisches Präfix interpretiert, das mit dem Anfang des zweiten Teils des vollständig qualifizierten Namens der Partner-LUs übereinstimmen muss. Der generische Eintrag **A.F** würde beispielsweise mit den Partner-LU-Namen **A.FRED** oder **A.FREDDY** (nicht jedoch mit **APPN.FRED** oder **A.B**) übereinstimmen.

Wenn Sie für beide Zeichenfolgen keinen Wert angeben, stimmt die generische Partner-LU-Definition mit allen Namen von Partner-LUs überein.

### Namen der Partner-LU generisch verwenden

Information, ob der Name der Partner-LU als Platzhalter oder buchstäblich als vollständig qualifizierter LU-Name verwendet werden soll.

## Zusätzliche Konfiguration

Führen Sie nach dem Konfigurieren der Partner-LU die folgenden Konfigurationsschritte aus:

- Wie Sie ein aufrufbares Transaktionsprogramm definieren, können Sie im Abschnitt „TPs definieren“ nachlesen.
- Wie Sie einen Modus definieren, können Sie im Abschnitt „Modi und Serviceklassen definieren“ auf Seite 100 nachlesen.
- Wie Sie CPI-C-Nebeninformationen definieren, können Sie im Abschnitt „CPI-C-Nebeninformationen definieren“ auf Seite 105 nachlesen.
- Wie Sie APPC-Sicherheit definieren, können Sie im Abschnitt „APPC-Sicherheit konfigurieren“ auf Seite 108 nachlesen.
- Wie Sie die 5250-Kommunikation konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.

---

## TPs definieren

In diesem Abschnitt wird das Definieren eines APPC-Transaktionsprogramms erläutert.

In den meisten Fällen müssen Sie TPs, die auf dem CS-Linux-System ausgeführt werden, nicht definieren. In folgenden Fällen ist jedoch eine TP-Definition erforderlich:

### APPC-Kenndaten

Wenn das TP auf dem CS-Linux-Computer das aufrufende TP (oder das den APPC-Dialog startende Quellen-TP) ist und Sie den Zugriff auf das TP nicht einschränken müssen, ist es nicht notwendig, das TP zu definieren. Sie können jedoch ein APPC-TP definieren (siehe Abschnitt „Parameter für TP-Definition“ auf Seite 99), um die folgenden Kenndaten anzugeben:

- Um Dialogsicherheit für das Transaktionsprogramm zu definieren.
- Um anzugeben, ob das Transaktionsprogramm den formatfreien Datenaustausch oder den Austausch formatierter Daten verwendet.
- Um die Synchronisationspunktverarbeitung anzugeben.
- Um die Bearbeitung von PIP-Daten anzugeben.

### Aufrufbare TPs

Wenn das Transaktionsprogramm als Reaktion auf eine ankommende Zuordnungsanforderung automatisch gestartet werden soll, müssen Sie das Programm als aufrufbares TP definieren. Gehen Sie dabei wie im Abschnitt „Aufrufparameter für Transaktionsprogramme auf einem Server“ auf Seite 96 beschrieben vor.

Ein aufrufbares Transaktionsprogramm (oder Ziel-TP) ist ein TP, das in Reaktion auf eine ankommende Zuordnungsanforderung gestartet wird. Für ein aufrufbares Transaktionsprogramm müssen Sie eine TP-Definition erstellen. Ein aufrufbares TP kann ein APPC-Transaktionsprogramm sein, das das Verb RECEIVE\_ALLOCATE absetzt, oder eine CPI-C-Anwendung, die den Aufruf Accept\_Conversation oder Accept\_Incoming absetzt.

**Anmerkung:** Im vorliegenden Handbuch werden diese drei API-Aufrufe unter der Bezeichnung „Receive\_Allocate“ zusammengefasst.

Sie können ein aufrufbares TP auch so definieren, dass ankommende Zuordnungsanforderungen an ein aktives TP weitergeleitet werden

Für ein aufrufbares Transaktionsprogramm können Sie auch ein Zeitlimit angeben, um die Wartezeit für eine Zuordnungsanforderung zu begrenzen. (Diese Option können Sie nur im Befehlszeilenverwaltungsprogramm konfigurieren.)

Communications Server für Linux benutzt die Definition aufrufbarer Transaktionsprogramme für folgende Zwecke:

- Wenn ein Transaktionsprogramm einen Aufruf Receive\_Allocate absetzt, sucht Communications Server für Linux nach einer Definition für ein aufrufbares Transaktionsprogramm mit dem entsprechenden TP-Namen. Sollte die Definition existieren und ein Zeitlimit für Receive\_Allocate enthalten, verwendet Communications Server für Linux diesen Wert beim Verarbeiten des Aufrufs Receive\_Allocate. Andernfalls wird die Standardeinstellung verwendet (kein Zeitlimit, so dass das Transaktionsprogramm unendlich lange wartet).
- Wenn auf dem Zielsystem eine Allocate-Anforderung ankommt und das angeforderte TP nicht bereits durch einen anstehenden Aufruf Receive\_Allocate aktiviert ist, sucht Communications Server für Linux nach einer TP-Definition mit dem im ankommenden Zuordnungsaufruf enthaltenen TP-Namen. Sollte die Definition existieren, verwendet Communications Server für Linux die Informationen in dieser Definition zum Starten des Transaktionsprogramms (sofern mehrere Instanzen zulässig sind oder das TP noch nicht aktiviert ist) oder um festzulegen, dass die

ankommende Allocate-Anforderung in eine Warteschlange gestellt werden soll (sofern das Transaktionsprogramm bereits läuft und mehrere Instanzen nicht zulässig sind).

Bei Bedarf können Sie für ein Transaktionsprogramm beide Arten von Definitionen konfigurieren (um beispielsweise Dialogsicherheit für ein aufrufbares Transaktionsprogramm zu definieren).

Verwenden Sie für das Konfigurieren einer TP-Definition eine der folgenden Methoden:

#### Zum Definieren von APPC-Kenndaten:

Wählen Sie eine der folgenden Methoden aus:

##### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Option **Transaktion** aus. Wenn Communications Server für Linux das TP-Fenster anzeigt, wählen Sie das untere Teilfenster aus, und klicken Sie auf die Schaltfläche **Neu**, oder wählen Sie eine vorhandene TP-Definition aus, und klicken Sie auf die Schaltfläche **Merkmale**.

##### Befehlszeilenverwaltungsprogramm

Setzen Sie den Befehl **snaadmin define\_tp** ab.

#### Zum Definieren eines aufrufbaren Transaktionsprogramms:

Die Konfigurationsmethoden für Server und Clients unterscheiden sich:

- Verwenden Sie auf einem Server eine der folgenden Methoden:

##### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Option **Transaktion** aus. Wenn Communications Server für Linux das TP-Fenster anzeigt, wählen Sie das obere Teilfenster aus, und klicken Sie auf die Schaltfläche **Neu**, oder wählen Sie eine vorhandene Definition für ein aufrufbares TP aus, und klicken Sie auf die Schaltfläche **Merkmale**.

##### Befehlszeilenverwaltungsprogramm

Setzen Sie den Befehl **snatpinstall** ab.

•

UNIX

Setzen Sie auf einem IBM Remote API Client unter AIX oder Linux den Befehl **snatpinstall** ab.

•

WINDOWS

Setzen Sie auf einem Windows-Client den Befehl **tpinst32** ab. (Dieser Befehl gilt für 32-Bit- und x64-Versionen von Windows.)



Informationen zur Verwendung des Befehls **snatpinstall** bzw. **tpinst32** finden Sie in Anhang B, „Aufrufbares TP mit Befehlszeilendienstprogramm konfigurieren“, auf Seite 189.

### Aufrufparameter für Transaktionsprogramme auf einem Server

Dieser Abschnitt beschreibt die Parameter, die das Motif-Verwaltungsprogramm oder das Befehlszeilenverwaltungsprogramm benötigt, um ein aufrufbares TP auf einem Server zu konfigurieren. Hinweise zum Konfigurieren eines aufrufbaren TP auf einem Client finden Sie in Anhang B, „Aufrufbares TP mit Befehlszeilendienstprogramm konfigurieren“, auf Seite 189.

Für ein Transaktionsprogramm, das auf dem lokalen Knoten aufgerufen werden kann, sind die folgenden Parameter erforderlich:

#### *Name des Transaktionsprogramms*

Der Name eines Transaktionsprogramms in einem der folgenden Formate:

##### **Anwendungs-TP**

Wenn das Transaktionsprogramm eine Benutzeranwendung ist, geben Sie den Namen in normalen Zeichen (max. 64) an.

##### **Service-TP**

Wenn das Transaktionsprogramm ein SNA-Service-TP ist, geben Sie den Namen im Hexadezimalformat (bis zu acht Hexadezimalziffern, die 4 Byte repräsentieren) ein.

Sie können mehrere mit APPC aufrufbare Transaktionsprogramme mit demselben Namen definieren, sofern jede TP-Definition einen anderen LU-Aliasnamen enthält. Für TPs, die mit CPI-C aufgerufen werden können, ist dies nicht möglich, da Sie keinen bestimmten LU-Aliasnamen zur Verwendung angeben können. Jedes mit CPI-C aufrufbare TP muss einen anderen Namen haben.

#### *Mit den Parametern kann jede LU bzw. eine spezifische LU aufgerufen werden*

Wenn das TP ein APPC-Transaktionsprogramm ist, gibt dieser Parameter an, ob das Transaktionsprogramm auf jeder beliebigen oder nur einer bestimmten LU aufrufbar sein soll. Standardmäßig kann das Transaktionsprogramm auf jeder LU aufgerufen werden.

**Anmerkung:** Ist das TP eine CPI-C-Anwendung, muss dieses Feld so gesetzt werden, dass das TP auf jeder LU aufgerufen werden kann. CPI-C kann keine ankommenden Attach-Anforderungen von einer bestimmten LU akzeptieren. Wenn Sie versuchen, diese Option für eine CPI-C-Anwendung anzugeben, führt dies beim Routing der ankommenden Attach-Anforderung an das TP zu Fehlern.

#### *LU-Aliasname*

Dieses Feld darf nicht verwendet werden, wenn das TP eine CPI-C-Anwendung ist. Ist das TP eine APPC-Anwendung, wird es nur verwendet, wenn Sie angeben, dass mit den Parametern für diese TP-Definition jede LU aufgerufen werden kann.

Der Aliasname der lokalen LU, von der das Transaktionsprogramm ankommende Attach-Anforderungen akzeptieren soll. Dieser Name muss mit dem Namen einer lokalen APPC-LU auf dem CS-Linux-Knoten übereinstimmen. Wenn Sie keinen LU-Aliasnamen angeben, akzeptiert das Transaktionsprogramm Attach-Anforderungen von allen lokalen LUs.

Wird ein nicht aus Leerzeichen bestehender LU-Aliasname angegeben, muss das TP die erweiterte Form des Verbs RECEIVE\_ALLOCATE verwenden und diesen LU-Aliasnamen als Parameter für das Verb angeben. Dies ermöglicht Communications Server für Linux, die ankommende Attach-An-

forderung an das richtige TP weiterzuleiten. Informationen zu den verschiedenen Formaten des Verbs RECEIVE\_ALLOCATE finden Sie im *Communications Server for Linux APPC Programmer's Guide*. Wenn das Transaktionsprogramm den richtigen LU-Aliasnamen nicht in die Anwendung integrieren, sondern zur Laufzeit bestimmen soll, können Sie eine Umgebungsvariable (mit dem Parameter *Umgebung*) auf den entsprechenden LU-Aliasnamen setzen und die Anwendung so konfigurieren, dass anhand dieser Umgebungsvariablen bestimmt wird, wie das Verb RECEIVE\_ALLOCATE abgesetzt werden soll.

Sie können mehrere Transaktionsprogramme mit demselben Namen definieren, sofern jede TP-Definition einen anderen LU-Aliasnamen enthält.

#### *Ankommende Zuordnungen in Warteschlange stellen*

Wenn Sie diese Option auswählen, ist das Transaktionsprogramm ein TP mit Warteschlangenbetrieb. Alle Allocate-Anforderungen, die ankommen, während das Transaktionsprogramm läuft, werden in eine Warteschlange gestellt, bis das Transaktionsprogramm einen weiteren Aufruf Receive\_Allocate absetzt oder beendet wird und neu gestartet werden kann. Eine ankommende Allocate-Anforderung wird nur dann an dieses Transaktionsprogramm weitergeleitet, wenn sie von einer LU empfangen wird, die für das Routing von Allocate-Anforderungen an diesen Computer konfiguriert ist, oder wenn sie von einer LU auf diesem Computer empfangen wird, für die keine Routing-Daten konfiguriert sind.

Wenn Sie diese Option nicht auswählen, ist das Transaktionsprogramm ein TP ohne Warteschlangenbetrieb. Communications Server für Linux startet bei jeder für das TP ankommenden Allocate-Anforderung eine neue Kopie des Transaktionsprogramms. Ein TP ohne Warteschlangenbetrieb kann nicht von einem Bediener gestartet werden. Es wird immer automatisch von Communications Server für Linux gestartet. Für ein TP ohne Warteschlangenbetrieb lässt Communications Server für Linux die gleichzeitige Ausführung mehrerer Kopien zu. Alle Kopien werden gemäß der Definition durch die Parameter *Benutzer-ID* und *Gruppen-ID* mit derselben Benutzer- und Gruppen-ID und im selben Arbeitsverzeichnis ausgeführt. Sollte das Transaktionsprogramm in Dateien auf dem lokalen System schreiben, müssen Sie sicherstellen, dass verschiedene Kopien des Transaktionsprogramms nicht gegenseitig ihre Dateien überschreiben.

Wenn ein TP ohne Warteschlangenbetrieb einen Dialog beendet hat, kann es beendet werden oder ein weiteres Verb RECEIVE\_ALLOCATE absetzen. Für häufig verwendete Programme bietet dies eine Möglichkeit, den zum Starten einer neuen Programminstanz für jeden Dialog erforderlichen Systemaufwand zu vermeiden. Jedes Mal, wenn eine Attach-Anforderung für ein automatisch gestartetes TP ohne Warteschlangenbetrieb empfangen wird, überprüft Communications Server für Linux, ob bereits ein RECEIVE\_ALLOCATE von einer Instanz dieses TP ansteht. Ist dies der Fall, wird das TP für den ankommenden Dialog verwendet. Andernfalls startet Communications Server für Linux eine neue Instanz des Programms.

#### *Ankommende Allocate-Anforderungen an aktives TP weiterleiten*

Diese Option wird nur angewendet, wenn mehrere Instanzen nicht unterstützt werden.

Wählen Sie diese Option aus, wenn das Transaktionsprogramm ein TP mit Broadcast- und Warteschlangenbetrieb ist. Alle Allocate-Anforderungen, die ankommen, während das Transaktionsprogramm läuft, werden in eine Warteschlange gestellt, bis das Transaktionsprogramm einen weiteren Aufruf Receive\_Allocate absetzt oder beendet wird und neu gestartet werden

## TPs definieren

kann. Nach dem Start des TP werden Informationen zum TP per Broadcast an alle Server im LAN gesendet. Wenn eine LU auf einem anderen Computer, für die keine Routing-Informationen konfiguriert sind, eine ankommende Allocate-Anforderung empfängt, kann sie das TP dynamisch lokalisieren und die Allocate-Anforderung an das TP weiterleiten.

Bei Verwendung dieser Option müssen keine expliziten Routing-Informationen für LUs konfiguriert werden. Außerdem kann die Arbeitslast durch Ausführung mehrerer Kopien eines TP auf verschiedenen Computern verteilt werden. Falls Sie jedoch die Informationsübermittlung per Broadcast verhindern möchten, um den Datenverkehr im LAN zu reduzieren, oder falls Sie sicherstellen müssen, dass bei einer bestimmten LU ankommende Allocate-Anforderungen immer an dieselbe Kopie des TP weitergeleitet werden, sollten Sie diese Option nicht auswählen.

### *Vollständiger Pfad zur ausführbaren Datei des Transaktionsprogramms*

Der vollständige Pfad- und Dateiname der ausführbaren Datei dieses Transaktionsprogramms.

Für die Datei muss eine Ausführungsberechtigung für den durch den Parameter *Benutzer-ID* angegebenen Benutzer vorliegen. Soll die ausführbare Datei mit einer auf root gesetzten *Benutzer-ID* ausgeführt werden, muss root der Eigner der Datei sein und über die Berechtigungen *setuid* und *setgid* verfügen, damit die Datei von Communications Server für Linux automatisch gestartet werden kann.

### *Argumente*

Alle an das Transaktionsprogramm zu übergebenden und durch Leerzeichen voneinander getrennten Argumente. Die Argumente werden in der hier angegebenen Reihenfolge an das Transaktionsprogramm übergeben.

Dieser Wert ist optional. Wenn sie nicht angegeben wird, wird das Transaktionsprogramm ohne Befehlszeilenargumente aufgerufen.

### *Benutzer-ID*

Die Benutzer-ID, die Communications Server für Linux zum Starten des Transaktionsprogramms verwendet. Diese Zeile ist erforderlich und muss angegeben werden. Die ID muss eine gültige Linux-Anmelde-ID auf dem CS-Linux-Computer sein.

Das Transaktionsprogramm wird in dem dieser Benutzer-ID zugeordneten Benutzerverzeichnis gestartet. Dieses Benutzerverzeichnis ist außerdem der Standardpfad für Trace-Dateien und andere Dateien, auf die das Transaktionsprogramm zugreift (sofern er nicht von der Anwendung mit einem vollständigen Pfad überschrieben wird). Wenn die Anwendung einen Dateinamen ohne Pfad angibt, sucht Communications Server für Linux in diesem Benutzerverzeichnis nach der Datei. Gibt die Anwendung einen Dateinamen mit einem relativen Pfad an, sucht Communications Server für Linux ausgehend von diesem Benutzerverzeichnis im angegebenen relativen Verzeichnis nach der Datei.

Der angegebene Benutzer muss für die vom Parameter *Vollständiger Pfad zur ausführbaren Datei des Transaktionsprogramms* angegebene ausführbare Datei die Ausführungsberechtigung haben. Ist die *Benutzer-ID* auf root gesetzt, muss root der Eigner der Datei sein und über die Berechtigungen *setuid* und *setgid* verfügen, damit die Datei von Communications Server für Linux automatisch gestartet werden kann.

### *Gruppen-ID*

Die Gruppen-ID, die Communications Server für Linux zum Starten des

Transaktionsprogramms verwendet. Die ID muss eine gültige Linux-Gruppen-ID auf dem CS-Linux-Computer sein.

Dieser Parameter ist optional. Wird er nicht aufgenommen, wird der Standardwert `sna` verwendet.

#### *Standardeingabe*

Geben Sie den vollständigen Pfadnamen der vom TP verwendeten Standardeingabedatei oder -einheit an.

Dieser Parameter ist optional. Wird er nicht aufgenommen, wird der Standardwert `/dev/null` verwendet.

#### *Standardausgabe*

Geben Sie den vollständigen Pfadnamen der vom TP verwendeten Standardausgabedatei oder -einheit an.

Dieser Parameter ist optional. Wird er nicht aufgenommen, wird der Standardwert `/dev/null` verwendet.

#### *Standardfehler*

Geben Sie den vollständigen Pfadnamen der vom TP verwendeten Standardfehlerdatei oder -einheit an.

Dieser Parameter ist optional. Wird er nicht aufgenommen, wird der Standardwert `/dev/null` verwendet.

#### *Umgebung*

Geben Sie alle für das Transaktionsprogramm erforderlichen Umgebungsvariablen an.

Jede Variable wird im Format *Umgebungsvariable=Wert* angegeben und kann bis zu 255 Zeichen umfassen. Die Zeichenfolge *Umgebungsvariable=Wert* darf vor und nach dem Zeichen `=` keine Tabulatorzeichen oder Leerzeichen enthalten.

Falls Sie im Motif-Verwaltungsprogramm mehr als eine Umgebungsvariable (bis maximal 64 Variablen) angeben müssen, trennen Sie diese durch das Zeichen `|`. Die Variablen werden in der hier angegebenen Reihenfolge gesetzt.

Ist das TP eine CPI-C-Anwendung, können Sie dieses Feld nicht zum Setzen der Umgebungsvariablen `APPCLU` verwenden. Für eine automatisch geladene CPI-C-Anwendung kann die lokale LU nicht angegeben werden.

Dieses Feld ist optional. Wird es nicht aufgenommen, werden keine Umgebungsvariablen verwendet.

## Parameter für TP-Definition

Sie können ein APPC-Transaktionsprogramm konfigurieren, um Dialogsicherheit, Dialogart, Synchronisationsebene und die Bearbeitung von PIP-Daten anzugeben. Für das Definieren eines Transaktionsprogramms für die APPC-Kommunikation sind die folgenden Parameter erforderlich:

#### *Name des Transaktionsprogramms*

Der Name eines Transaktionsprogramms in einem der folgenden Formate:

##### **Anwendungs-TP**

Wenn das Transaktionsprogramm eine Benutzeranwendung ist, geben Sie den Namen in normalen Zeichen (max. 64) an.

##### **Service-TP**

Wenn das Transaktionsprogramm ein SNA-Service-TP ist, geben Sie

## TPs definieren

den Namen im Hexadezimalformat (bis zu acht Hexadezimalziffern, die 4 Byte repräsentieren) an.

### *Sicherheit auf Dialogebene erforderlich*

Wählen Sie diese Option aus, wenn eine Zuordnungsanforderung einen gültigen Benutzernamen und ein Kennwort enthalten muss (oder eine Anzeige, dass das Kennwort bereits geprüft wurde). Wenn Sie diese Option nicht auswählen, ist keine Prüfung erforderlich.

### *Zugriff einschränken*

Wählen Sie diese Option aus, wenn der Benutzername in eine Sicherheitszugriffsliste aufgenommen werden muss. Dieses Feld findet nur Anwendung, wenn die Option *Sicherheit auf Dialogebene erforderlich* ausgewählt wurde.

### *Sicherheitszugriffsliste*

Name einer Sicherheitszugriffsliste mit Benutzer-IDs, die berechtigt sind, auf dieses Transaktionsprogramm zuzugreifen. Wenn die Option *Zugriff einschränken* ausgewählt wurde, müssen Sie diesen Wert angeben.

### *Dialogart*

Geben Sie an, ob das TP nur Dialoge mit dem Austausch formatierter Daten, nur Dialoge mit formatfreiem Datenaustausch oder beide Arten von Dialogen akzeptiert.

### *Synchronisationsebene*

Geben Sie die vom TP akzeptierten Ebenen für die Bestätigung der Synchronisation an. Weitere Informationen zur Bestätigung der Synchronisation finden Sie im "Communications Server for Linux APPC Programmer' Guide". Wählen Sie einen der folgenden Werte aus:

- Keine
- Bestätigen
- Synchronisationspunkt
- Keine oder Bestätigen
- Keine, Bestätigen oder Synchronisationspunkt

### *PIP erlaubt*

Wählen Sie diese Option aus, wenn das TP PIP-Daten (Programinitialisierungsparameter) akzeptiert.

---

## Modi und Serviceklassen definieren

Ein Modus gibt eine Reihe von Kenndaten an, die eine lokale LU (des Typs 6.2) für die Kommunikation mit ihrer Partner-LU verwendet. Zu diesen Kenndaten gehören Informationen zur Art der Datenübertragung zwischen den beiden LUs (z. B. die maximale RU-Größe und die Größe des Nachrichtendosierfensters) und zur Fähigkeit der LUs, Parallelsitzungen aufzubauen.

Darüber hinaus müssen Sie ggf. die Voraussetzungen für den Kommunikationspfad zwischen den LUs angeben, z. B. das Erzwingen einer bestimmten Stufe der Netzsicherheit, die Minimierung der Übertragungszeit oder die Vermeidung der Verwendung teurer DFV-Verbindungen. Sie können diese Anforderungen mit einer Serviceklasse (COS) definieren, die für Merkmale wie Transaktionszeit, Übertragungskosten und Netzsicherheit zulässige Mindest- und Höchstwerte angibt. Die COS gewichtet außerdem verschiedene Bereiche dieser Werte. Auf dieser Grundlage kann der Knoten die beste Route durch das Netz berechnen, wenn zwei oder mehr Routen zu derselben fernen LU verfügbar sind.

Wenn der CS-Linux-Knoten ein Netzknoten ist, enthält die Definition jedes Modus den Namen der für diesen Modus erforderlichen COS. Wenn der CS-Linux-Knoten ein LEN-Knoten oder ein Endknoten ist, müssen Sie dem Modus keine Serviceklasse zuordnen. Der Name der Serviceklasse wird in diesem Fall dynamisch bestimmt.

SNA definiert eine Reihe von Standardmodi und zugeordneten Serviceklassen, die den Anforderungen der meisten Systeme genügen. In der Regel müssen Sie keine zusätzlichen Modi und Serviceklassen definieren. Einen Modus müssen Sie nur definieren, wenn der erforderliche Modus keiner der im Modusfenster angezeigten Standardmodi ist.

Wird der Modusname in einem ankommenden Dialog nicht erkannt, wird der Standardmodus verwendet. Sollten Sie keinen Standardmodus angeben, wird der leere Modusname als Standardmodus verwendet.

Tabelle 2 gibt einen Überblick über die Namen der Standardmodi und der zugehörigen Serviceklassen. Weitere Informationen zu den diesen Standardnamen zugeordneten Parametern finden Sie in den IBM SNA-Handbüchern *LU 6.2 Reference—Peer Protocols* (für Modi) und *APPN Architecture Reference* (für Serviceklassen).

*Tabelle 2. Namen der Standardmodi und Serviceklassen*

Modusname	Zugeordneter COS-Name	Anwendung
(Leername)	#CONNECT	Sitzungen, die keinen Modusnamen angeben (grundlegende COS-Standardparameter)
#BATCH	#BATCH	Von Anwendungen mit Stapelverarbeitung benutzte Sitzungen
#BATCHSC	#BATCHSC	Von Anwendungen mit Stapelverarbeitung und minimaler Sicherheitsstufe für das Routing benutzte Sitzungen
#BATCHC	#BATCH	Sitzungen, die in Anwendungen mit Stapelverarbeitung die Komprimierung verwenden
#BATCHCS	#BATCH	Sitzungen, die in Anwendungen mit Stapelverarbeitung und minimaler Sicherheitsstufe für das Routing die Komprimierung verwenden
#INTER	#INTER	Von interaktiven Anwendungen benutzte Sitzungen
#INTERSC	#INTERSC	Von interaktiven Anwendungen mit minimaler Sicherheitsstufe für das Routing benutzte Sitzungen
#INTERC	#INTER	Sitzungen, die in interaktiven Anwendungen die Komprimierung verwenden
#INTERCS	#INTER	Sitzungen, die in interaktiven Anwendungen mit minimaler Sicherheitsstufe für das Routing die Komprimierung verwenden
SNASVCMG	SNASVCMG	CNOS-Sitzungen (Change Number Of Sessions) und MS-Sitzungen (Management Services)
CPSVCMG	CPSVCMG	CP-CP-Sitzungen zwischen Knoten
CPSVRMGR	CPSVRMGR	Für DLUR (Dependent LU Requester) benutzte CP-CP-Sitzungen
QPCSUPP	#CONNECT	Für die 5250-Emulation benutzte Sitzungen

Ein einmal konfigurierter Modus kann von jeder APPC- oder CPI-C-Anwendung zum Aktivieren einer Sitzung zwischen einer lokalen LU und einer Partner-LU ver-

## Modi und Serviceklassen definieren

wendet werden. Eine APPC-Anwendung muss angeben, welcher Modus verwendet werden soll. Eine CPI-C-Anwendung kann hingegen die CPI-C-Nebeninformationen benutzen (die den Modusnamen enthalten). Weitere Informationen zum Konfigurieren von CPI-C-Nebeninformationen finden Sie im Abschnitt „CPI-C-Nebeninformationen definieren“ auf Seite 105.

Verwenden Sie für das Konfigurieren eines Modus oder einer Serviceklasse eine der folgenden Methoden:

### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Option **Modi** aus. Wählen Sie anschließend im Modusfenster die Option **Neu** aus.

### Befehlszeilenverwaltungsprogramm

Setzen Sie zum Definieren eines Modus den folgenden Befehl ab:

```
define_mode
```

Setzen Sie zum Ändern des Standardmodus den folgenden Befehl ab:

```
define_defaults
```

Setzen Sie zum Definieren einer Serviceklasse den folgenden Befehl ab:

```
define_cos
```

## Konfigurationsparameter für Modi

Für die Moduskonfiguration sind die folgenden Parameter erforderlich:

*Name* Der Name des Modus, den Sie definieren. Der Modusname ist eine aus 1–8 Zeichen bestehende Zeichenfolge.

APPC-Anwendungen, die mit diesem Modus arbeiten, können diesen Namen ebenfalls verwenden. Dies gilt sowohl für lokale als auch für ferne Anwendungen. Stimmen Sie den Namen deshalb mit dem Anwendungsentwickler ab (bzw. lesen Sie für Anwendungen eines anderen Herstellers die entsprechenden Informationen in der Produktdokumentation).

*Serviceklassenname*

Der Name der Serviceklasse für diesen Modus. Der Name ist eine aus 1–8 Zeichen bestehende Zeichenfolge. Für Modi, die für den interaktiven Datenaustausch benutzt werden, können Sie in der Regel **#INTER** angeben, für Modi, die für die Übertragung von Massendaten benutzt werden, **#BATCH**.

Dieses Feld findet nur für Netzknoten Anwendung.

Sollten Sie nicht wissen, welchen Wert Sie angeben müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

*Sitzungsbegrenzungen*

Verwenden Sie zum Angeben der Sitzungsbegrenzungen die folgenden Felder:

*Einleitende Sitzungsbegrenzung*

Die Höchstanzahl Sitzungen (bis zur maximalen Sitzungsbegrenzung), die ein LU-Paar in diesem Modus ausführen kann, solange nicht mit CNOS ein anderer Höchstwert ausgehandelt wird.

In diesem Feld wird normalerweise der Wert 8 angegeben. Sollten Sie nicht sicher sein, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz oder den Entwickler der APPC-Anwendung (bzw. lesen Sie für Anwendungen eines anderen Herstellers die entsprechenden Informationen in der Produktdokumentation).

### *Maximale Sitzungsbegrenzung*

Die für ein LU-Paar zulässige Höchstzahl von Sitzungen (bis zu 32.767) in diesem Modus. Dieser Wert ist auch bei CNOS-Vereinbarung verbindlich.

Dieses Feld wird in der Regel auf denselben Wert gesetzt wie die einleitende Sitzungsbegrenzung. Sollten Sie nicht sicher sein, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz oder den Entwickler der APPC-Anwendung (bzw. lesen Sie für Anwendungen eines anderen Herstellers die entsprechenden Informationen in der Produktdokumentation).

### *Mindestanzahl Sitzungen für Konfliktgewinner*

Die Anzahl Sitzungen (bis zur Sitzungsbegrenzung), die Communications Server für Linux für die lokale LU als Konfliktgewinner reservieren muss.

Eine für die meisten Fälle geeignete Angabe für dieses Feld ist 0. Sollten Sie jedoch nicht sicher sein, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Die Summe aus der Mindestanzahl Sitzungen für Konfliktgewinner und der Mindestanzahl Sitzungen für Konfliktverlierer darf den Wert für die einleitende Sitzungsbegrenzung nicht überschreiten.

### *Mindestanzahl Sitzungen für Konfliktverlierer*

Die Anzahl Sitzungen, die Communications Server für Linux für die lokale LU als Konfliktverlierer reservieren muss. Dieser Wert legt zusammen mit dem Wert im Feld *Mindestanzahl Sitzungen für Konfliktgewinner* fest, wie der Konflikt für eine Sitzung gelöst wird.

Eine für die meisten Fälle geeignete Angabe ist 0. Sollten Sie jedoch nicht sicher sein, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Die Summe aus der Mindestanzahl Sitzungen für Konfliktgewinner und der Mindestanzahl Sitzungen für Konfliktverlierer darf den Wert für die einleitende Sitzungsbegrenzung nicht überschreiten.

### *Automatisch aktivierte Sitzungen*

Die Anzahl Sitzungen (bis zum Wert für die Mindestanzahl Sitzungen für Konfliktgewinner), die automatisch aktiviert werden, nachdem für eine Sitzung zwischen einer lokalen LU und einer Partner-LU in diesem Modus eine CNOS-Vereinbarung stattgefunden hat. Wenn Sie in diesem Feld einen Wert angeben, kann eine LU, die diesen Modus benutzt, automatisch Sitzungen starten, sobald ein Transaktionsprogramm die sofortige Zuordnung eines Dialogs anfordert.

### *Fenster für Empfangsnachrichtendosierung*

Geben Sie in den folgenden Feldern an, wie viele RUs vor dem Senden einer SNA-Antwort auf die Nachrichtendosierung empfangen werden können:

## Modi und Serviceklassen definieren

### *Anfangswert*

Der Anfangswert für die Anzahl Anforderungseinheiten (RUs, Request Units), die die lokale LU empfangen kann, bevor Sie eine Antwort auf die Nachrichtendosierung an die ferne LU senden muss. Eine sichere Einstellung für dieses Feld ist 4.

Ein höherer Wert kann unter bestimmten Umständen den Durchsatz verbessern, zieht aber auch eine höhere Speicherauslastung nach sich.

### *Maximum*

Die maximale Anzahl Anforderungseinheiten (RUs, Request Units), die die lokale LU empfangen kann, bevor Sie eine Antwort auf die Nachrichtendosierung an die ferne LU senden muss.

Dieser Wert ist optional. Wird er nicht angegeben, ist die Größe des Fensters für Empfangsnachrichtendosierung uneingeschränkt. Wenn ein Wert angegeben wird, wird er verwendet, um bei angepasster Nachrichtendosierung die Größe des Fensters für Empfangsnachrichtendosierung zu begrenzen. Wird nicht mit angepasster Nachrichtendosierung gearbeitet, wird dieser Wert ignoriert.

Das Fenster für Nachrichtendosierung kann zwischen 0 und 32767 Bytes groß sein. Der Wert 0 gibt ein unbegrenztes Fenster an.

Wenn der Nachbarknoten nur die statische Nachrichtendosierung unterstützt, legen diese Werte die Größe des Fensters für statische Nachrichtendosierung fest. Der Nachbarknoten kann unabhängig davon noch immer eine Fenstergröße aushandeln. Wenn der Nachbarknoten die angepasste Nachrichtendosierung verwendet, legen diese Werte die Ausgangsfenstergröße fest.

### *Zeitlimit angeben*

Wählen Sie diese Option aus, um die Zeit in Sekunden (0 - 65535) anzugeben, die eine diesen Modus verwendende LU 6.2-Sitzung inaktiv sein muss, bevor sie unterbrochen wird. Eine Änderung dieses Wertes wirkt sich nur auf Sitzungen aus, für deren Aktivierung diese Definition relevant ist. (Bereits aktive Sitzungen sind nicht betroffen.)

Wenn Sie den Wert 0 angeben, wird für Sitzungen eine Zeitlimitüberschreitung festgestellt, sobald diese verfügbar werden.

### *Max. RU-Größe einschränken*

Wählen Sie diese Option aus, wenn Sie die maximale RU-Größe angeben möchten, die festlegt, wie viele Daten vor dem Senden an die Partner-LU im Puffer gesammelt werden.

Der obere Grenzwert kann zwischen 256 und 62440 Bytes liegen. Eine sichere Einstellung für den oberen Grenzwert sind 1024 Bytes. Ein höherer Wert kann unter bestimmten Umständen den Durchsatz verbessern, zieht aber auch eine höhere Speicherauslastung nach sich.

Die Untergrenze kann bei 0 liegen oder einen Wert haben, der zwischen 256 und dem angegebenen oberen Grenzwert liegt.

Unterscheidet sich der Wert in diesem Feld von der für den fernen Knoten definierten RU-Größe, kann die für eine Sitzung mit diesem Knoten verwendete Größe ausgehandelt werden, um für die Sitzung eine angemessene RU-Größe zu erzielen. Der tatsächliche Wert kann nicht kleiner als der Wert im Feld für die Untergrenze sein.

Mit Hilfe dieser Werte und der Werte für die Sende- und Empfangsnachrichtendosierung kann der Durchsatz auf Sitzungsebene zwischen der lokalen LU und der Partner-LU optimiert werden. Sollten Sie nicht wissen, welche Werte Sie verwenden sollen, beginnen Sie mit den Standardwerten und passen Sie die Werte dann entsprechend den Erfordernissen für eine Maximierung des Durchsatzes an.

### *Komprimierung unterstützt*

Gibt an, ob für Sitzungen, die diesen Modus verwenden, die Datenkomprimierung unterstützt wird. Wenn Sie diese Option nicht festlegen, wird keine Komprimierung verwendet.

Ist diese Option festgelegt, können Sie die maximale Komprimierungsstufe für ankommende und für abgehende Daten angeben. Dies sind zwei gesonderte Optionen, so dass Sie für die beiden Richtungen verschiedene Stufen angeben oder die Komprimierung nur in einer Richtung verwenden können. Für jede der beiden Richtungen können Sie für die Komprimierung keine auswählen oder einen der Werte RLE (minimale Komprimierung), LZ9 und LZ10 (maximale Komprimierung).

### **Auf SNA-definierte Werte zurücksetzen**

Wenn Sie einen Standardmodus in der Motif-Dialoganzeige ändern, können Sie auf diese Schaltfläche klicken, um die Werte der Modusparameter auf die SNA-definierten Werte zurückzusetzen.

## **Zusätzliche Konfiguration**

Führen Sie nach dem Konfigurieren des Modus die folgenden Konfigurationsaufgaben aus:

- Wie Sie CPI-C-Nebeninformationen definieren, können Sie im Abschnitt „CPI-C-Nebeninformationen definieren“ nachlesen.
- Wie Sie APPC-Sicherheit definieren, können Sie im Abschnitt „APPC-Sicherheit konfigurieren“ auf Seite 108 nachlesen.
- Wie Sie die 5250-Kommunikation konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.

---

## **CPI-C-Nebeninformationen definieren**

Wenn Ihr System eine CPI-C-Anwendung unterstützt, die symbolische CPI-C-Bestimmungsorte verwendet, müssen Sie die CPI-C-Nebeninformationen definieren. Die Nebeninformationen ordnen dem symbolischen Bestimmungsort Informationen zum Partner-TP, zur Partner-LU, zum Modus und zur Dialogsicherheit zu.

Den symbolischen Bestimmungsort für CPI-C kann Ihnen der Anwendungsentwickler nennen. (Für Anwendungen eines anderen Herstellers können Sie den Bestimmungsort der Produktdokumentation entnehmen.)

Verwenden Sie für das Konfigurieren von CPI-C-Nebeninformationen eine der folgenden Methoden:

### **Motif-Verwaltungsprogramm**

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Option **CPI-C** aus.

### **Befehlszeilenverwaltungsprogramm**

Geben Sie den folgenden Befehl ein:

```
define_cplic_side_info
```

### Konfigurationsparameter für CPI-C

Für jeden von der Anwendung verwendeten symbolischen CPI-C-Bestimmungsort müssen Sie die folgenden Informationen zusammenstellen:

*Name* Der von CPI-C-Anwendungen (Transaktionsprogrammen), die Sie ausführen wollen, verwendete Name des symbolischen Bestimmungsorts. Dieser Name kann 1–8 Zeichen lang sein.

Diesen Namen erfahren Sie beim Anwendungsentwickler (bzw. lesen Sie für Anwendungen eines anderen Herstellers die entsprechenden Informationen in der Produktdokumentation).

#### *Lokale LU*

Die lokale LU für alle Dialoge, die von Transaktionsprogrammen, die diese Nebeninformationen verwenden, über eine der folgenden Optionen eingeleitet werden:

#### *Aliasname der lokalen LU*

Ein Aliasname für eine lokale LU.

#### *Standard-LU verwenden*

Geben Sie diese Option an, wenn (bei vorhandenem Standardpool) eine LU des Standardpools oder (falls kein Standardpool definiert ist) die Steuerpunkt-LU des Knotens verwendet werden soll.

Wenn die Umgebungsvariable APPCLLU gesetzt ist, werden die von Ihnen bereitgestellten Informationen zur lokalen LU ignoriert. Stattdessen wird die für die Umgebungsvariable angegebene LU verwendet.

#### *Partner-LU*

Entweder ein Aliasname oder der vollständig qualifizierte Name der Partner-LU für Dialoge, die von lokalen Transaktionsprogrammen unter Verwendung dieser Nebeninformationen eingeleitet werden. Die Partner-LU muss eine LU sein, die auf dem Computer, der das Partner-TP ausführt, konfiguriert ist.

*Modus* Der Name des APPC-Modus, in dem auf die Partner-LU zugegriffen wird. In den meisten Fällen wird einer der folgenden vordefinierten Modi verwendet:

- Leername
- #BATCH
- #BATCHSC
- #INTER
- #INTERSC
- QPCSUPP

#### *Partner-TP*

Der Name des Transaktionsprogramms, mit dem die CPI-C-Anwendung kommuniziert.

- Wenn das Transaktionsprogramm eine Benutzeranwendung ist, geben Sie den Namen in normalen Zeichen (max. 64) an.
- Wenn das Transaktionsprogramm ein Service-TP ist, geben Sie den Namen im Hexadezimalformat (bis zu acht Hexadezimalziffern, die vier Byte repräsentieren) an.

Diesen Wert erfahren Sie beim Anwendungsentwickler (bzw. lesen Sie für Anwendungen eines anderen Herstellers die entsprechenden Informationen in der Produktdokumentation).

### *Sicherheit*

Die Stufe der Sicherheit auf Dialogebene, mit der Sie arbeiten wollen. Es gibt folgende Optionen:

**Keine** Das Partner-TP erfordert keine Überprüfung von Sicherheitsparametern.

#### **Dieselbe**

Das Partner-TP arbeitet mit Sicherheitseinrichtungen, akzeptiert aber das Ergebnis der Überprüfung der vom einleitenden Transaktionsprogramm angegebenen Werte für Benutzer-ID und Kennwort durch das lokale Transaktionsprogramm. Wenn Sie für die Sicherheitsstufe die Option **Dieselbe** auswählen, müssen Sie eine gültige Benutzer-ID angeben, die vom Partner-TP akzeptiert wird.

#### **Programm**

Das Partner-TP erfordert eine Benutzer-ID und ein Kennwort. Wenn Sie für die Sicherheitsstufe die Option **Programm** auswählen, müssen Sie eine gültige Benutzer-ID und ein gültiges Kennwort angeben, die vom Partner-TP akzeptiert werden.

#### **Programm, hohe Priorität**

Das Partner-TP erfordert eine Benutzer-ID und ein Kennwort. Der lokale und der ferne Knoten müssen Sicherheitsverbesserungen unterstützen, damit das Kennwort verschlüsselt wird.

Lesen Sie in der Dokumentation zur CPI-C-Anwendung nach, welche Sicherheitsparameter anzuwenden sind. Wenden Sie sich ggf. an den Anwendungsprogrammierer.

### *Benutzer-ID*

Wenn Sie für die Sicherheitsstufe die Option **Dieselbe**, **Programm** oder **Programm, hohe Priorität** ausgewählt haben, geben Sie eine Benutzer-ID an, die mit der einleitenden Nachricht an die ferne Anwendung gesendet werden soll. Der Wert muss mit einer Benutzer-ID übereinstimmen, die die Anwendung gemäß Definition akzeptiert.

Diese Benutzer-ID steht in keinerlei Bezug zu den Benutzer-IDs für die Linux-Anmeldung auf dem lokalen oder fernen Knoten. Wenn der ferne Knoten Communications Server für Linux ausführt, muss die Benutzer-ID über den Konfigurationsdialog für Dialogsicherheit auf dem fernen Knoten konfiguriert werden.

### *Kennwort*

Wenn Sie für die Sicherheitsstufe der Wert **Programm** oder **Programm, hohe Priorität** ausgewählt, geben Sie ein Kennwort an, das bei Zuordnung des Dialogs gesendet werden soll. Dieser Wert muss mit dem Kennwort übereinstimmen, das für die ferne Anwendung zum angegebenen Benutzernamen definiert wurde.

Dieses Kennwort steht in keinem Bezug zu Linux-Anmeldekennwörtern auf dem lokalen oder fernen Knoten. Wenn der ferne Knoten Communications Server für Linux ausführt, muss das Kennwort über den Konfigurationsdialog für Dialogsicherheit auf dem fernen Knoten konfiguriert werden.

### Zusätzliche Konfiguration

Führen Sie nach der CPI-C-Konfiguration die folgenden Konfigurationsschritte aus:

- Wie Sie APPC-Sicherheit definieren, können Sie im Abschnitt „APPC-Sicherheit konfigurieren“ nachlesen.
- Wie Sie die 5250-Kommunikation konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.

---

### APPC-Sicherheit konfigurieren

Für die APPC-Sicherheit können Sie die folgenden Konfigurationsschritte ausführen:

- Konfigurieren der Sitzungssicherheit, wie im Abschnitt „Sitzungssicherheit konfigurieren“ beschrieben.
- Konfigurieren der Dialogsicherheit, wie im Abschnitt „Dialogsicherheit konfigurieren“ auf Seite 109 beschrieben.
- Konfigurieren von Sicherheitszugriffslisten, wie im Abschnitt „Sicherheitszugriffsliste konfigurieren“ auf Seite 110 beschrieben.

### Sitzungssicherheit konfigurieren

Die Sicherheit auf Sitzungsebene wird für die Gültigkeitsüberprüfung von LU-LU-Sitzungen verwendet. Jede Definition umfasst den Namen einer lokalen LU, den Namen einer Partner-LU und ein Kennwort.

Communications Server für Linux überprüft die Gültigkeit von Sitzungen zwischen der lokalen LU und der Partner-LU mit Hilfe des Kennworts. (Die Kennwörter stehen in keinem Bezug zu Linux-Anmeldekennwörtern.)

Verwenden Sie für das Konfigurieren der Sitzungssicherheit eine der folgenden Methoden:

#### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Optionen **Sicherheit** und **Sicherheit auf Sitzungsebene** aus.

#### Befehlszeilenverwaltungsprogramm

Geben Sie den folgenden Befehl ein:

```
define_lu_lu_password
```

#### Konfigurationsparameter für Sitzungssicherheit

Für die Konfiguration der Sitzungssicherheit sind die folgenden Parameter erforderlich:

##### *Lokale LU*

Der LU-Name der lokalen LU. Der Name ist eine aus 1–8 Zeichen bestehende Zeichenfolge.

##### *Partner-LU*

Der vollständig qualifizierte LU-Name der Partner-LU.

### *Kennwort*

Ein Kennwort, das Communications Server für Linux zur Überprüfung der Gültigkeit von Sitzungen zwischen der lokalen LU und der Partner-LU verwendet. Das Kennwort ist eine 16-stellige Hexadezimalzahl, auf deren Grundlage ein Schlüssel erstellt wird, der beim Aufbauen der Sitzung ausgetauscht wird. Dieses Kennwort steht in keinem Bezug zu Linux-Anmeldekennwörtern auf dem lokalen oder fernen Knoten.

### **Zusätzliche Konfiguration**

Führen Sie nach dem Konfigurieren der Sitzungssicherheit die folgenden Konfigurationsaufgaben aus:

- Wie Sie die Dialogsicherheit konfigurieren, können Sie im Abschnitt „Dialogsicherheit konfigurieren“ nachlesen.
- Wie Sie die 5250-Kommunikation konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.

## **Dialogsicherheit konfigurieren**

Die Dialogsicherheit wird für die Überprüfung der Gültigkeit ankommender Dialoge verwendet. Jede Definition umfasst eine Benutzer-ID und ein Kennwort. Diese Benutzer-ID steht in keinerlei Bezug zu den Benutzer-IDs für die Linux-Anmeldung auf dem lokalen oder fernen Knoten.

Verwenden Sie für das Konfigurieren der Dialogsicherheit eine der folgenden Methoden:

### **Motif-Verwaltungsprogramm**

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Optionen **Sicherheit** und **Sicherheit auf Dialogebene** aus.

### **Befehlszeilenverwaltungsprogramm**

Geben Sie den folgenden Befehl ein:

```
define_userid_password
```

### **Konfigurationsparameter für Dialogsicherheit**

Für die Konfiguration der Dialogsicherheit sind die folgenden Parameter erforderlich:

#### *Benutzer-ID*

Die Benutzer-ID, die von einem fernen Knoten in einem ankommenden Dialog akzeptiert werden soll. Die Benutzer-ID kann bis zu zehn Zeichen lang sein.

#### *Kennwort*

Das Kennwort, das von einem fernen Knoten in einem ankommenden Dialog akzeptiert werden soll. Das Kennwort kann bis zu zehn Zeichen lang sein.

### **Zusätzliche Konfiguration**

Nach dem Konfigurieren der Dialogsicherheit können Sie die 5250-Kommunikation konfigurieren. Gehen Sie dabei wie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 beschrieben vor.

### Sicherheitszugriffsliste konfigurieren

Sie können eine APPC-Sicherheitszugriffsliste definieren, um den Zugriff auf eine LU oder ein Transaktionsprogramm (oder beides) zu steuern. Auf diese Liste kann eine Definition für eine lokale APPC-LU oder ein APPC-Transaktionsprogramm verweisen.

Verwenden Sie für das Konfigurieren einer Sicherheitszugriffsliste eine der folgenden Methoden:

#### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **APPC** und dann die Optionen **Sicherheit** und **Sicherheit auf Dialogebene** aus. Wählen Sie anschließend im Teilfenster für Sicherheitszugriffslisten die Option **Neu** aus.

#### Befehlszeilenverwaltungsprogramm

Geben Sie den folgenden Befehl ein:

```
define_security_access_list
```

### Konfigurationsparameter für Sicherheitszugriffsliste

Für die Konfiguration einer Sicherheitszugriffsliste sind die folgenden Parameter erforderlich:

*Name* Name der Sicherheitszugriffsliste. Die Definition für ein APPC-Transaktionsprogramm oder eine lokale APPC-LU kann mit diesem Namen auf die Zugriffsliste verweisen.

*Benutzer in Zugriffsliste*

Die Namen der in die Sicherheitszugriffsliste aufgenommenen Benutzer.

### Zusätzliche Konfiguration

Führen Sie nach dem Konfigurieren der Sicherheitszugriffsliste die folgenden Konfigurationaufgaben aus:

- Konfigurieren Sie den Zugriff auf Transaktionsprogramme wie im Abschnitt „TPs definieren“ auf Seite 93 beschrieben.

---

## Kapitel 7. Benutzeranwendungen konfigurieren

Dieses Kapitel enthält Anweisungen für das Konfigurieren von SNA-Ressourcen zur Unterstützung von Benutzeranwendungen, die mit der 3270-, 5250- und LUA-Kommunikation arbeiten. Die für solche Anwendungen erforderlichen SNA-Ressourcen sind LUs.

Für die 3270-, die LUA- und die abhängige APPC-Kommunikation müssen Sie abhängige LUs konfigurieren. Für die unabhängige APPC- und die 5250-Kommunikation können Sie die Standardsteuerpunkt-LU verwenden (die beim Konfigurieren des lokalen Knotens automatisch definiert wird) oder unabhängige LUs definieren.

Bevor Sie die in diesem Kapitel beschriebenen Ressourcen konfigurieren können, müssen Sie die folgenden Konfigurationsschritte ausführen:

- Konfigurieren Sie den Knoten wie im Abschnitt „Knoten konfigurieren“ auf Seite 58 beschrieben.
- Konfigurieren Sie die Konnektivität wie in Kapitel 4, „Konnektivitätskomponenten definieren“, auf Seite 63 beschrieben. Für die 3270-, die LUA- und die abhängige APPC-Kommunikation müssen Sie die Verbindung für die Unterstützung von abhängigem LU-Verkehr konfigurieren. Bei der unabhängigen APPC- und der 5250-Kommunikation muss die Verbindung unabhängigen LU-Verkehr unterstützen.

Wenn Sie übergeordnetes SNA Gateway oder DLUR verwenden, müssen Sie keine Direktverbindung zum Host konfigurieren. Weitere Informationen hierzu finden Sie im Abschnitt „SNA Gateway konfigurieren“ auf Seite 122 und im Abschnitt „DLUR-PUs definieren“ auf Seite 77.

Nachfolgend finden Sie eine Auflistung der für die einzelnen Benutzeranwendungen erforderlichen Konfigurationsschritte:

### 3270-Anwendungen

Konfigurieren Sie für die 3270-Kommunikation die folgenden Ressourcen:

1. Definieren Sie für ein 3270-Terminal oder einen 3270-Drucker eine abhängige LU. Gehen Sie dabei wie im Abschnitt „LUs der Typen 0–3 definieren“ auf Seite 81 beschrieben vor.
2. Sollen 3270-Terminals aus einem Pool von LUs auswählen können, müssen Sie einen LU-Pool definieren. Gehen Sie dabei wie im Abschnitt „LU-Pools definieren“ auf Seite 83 beschrieben vor. Verwendet ein Terminal eine dedizierte LU, können Sie diesen Schritt überspringen.

### 5250-Anwendungen

Konfigurieren Sie für die 5250-Kommunikation die folgenden Ressourcen:

1. Konfigurieren Sie den Knoten wie folgt für die APPC-Kommunikation:
  - a. Wenn Sie die Steuerpunkt-LU des lokalen Knotens verwenden können, müssen Sie keine lokale LU konfigurieren. Falls Sie die Definition einer lokalen LU benötigen (z. B., um die Sitzungssicherheit anwenden zu können), definieren Sie die lokale LU gemäß der Beschreibung im Abschnitt „Lokale LUs definieren“ auf Seite 86.
  - b. Wenn der lokale Knoten ein LEN-Knoten ist, müssen Sie die AS/400 gemäß der Beschreibung im Abschnitt „Ferne Knoten definieren“ auf Seite 88 als fernen Knoten definieren.

## Benutzeranwendungen konfigurieren

Ist der lokale Knoten ein APPN-Endknoten oder -Netzknoten, können Sie die Steuerpunkt-LU der AS/400 als Partner-LU verwenden und müssen keine weiteren Partner-LUs konfigurieren.

Sie müssen keine Modi definieren, da 5250 den Standardmodus QPCSUPP verwendet.

### LUA-Anwendungen

Konfigurieren Sie für die Unterstützung einer LUA-Anwendung die folgenden Ressourcen:

1. Definieren Sie gemäß der Beschreibung im Abschnitt „LUs der Typen 0–3 definieren“ auf Seite 81 eine abhängige LU.
2. Soll eine LUA-Anwendung aus einem Pool von LUs auswählen können, müssen Sie einen LU-Pool definieren. Gehen Sie dabei wie im Abschnitt „LU-Pools definieren“ auf Seite 83 beschrieben vor. Verwendet die Anwendung eine dedizierte LU, können Sie diesen Schritt überspringen.

Eine LUA-Anwendung benutzt für die Kommunikation mit einer Hostanwendung die Knotenressourcen (LUs der Typen 0-3). Sie müssen keine zusätzlichen Ressourcen definieren.

---

## Kapitel 8. Durchgriffsdienste konfigurieren

Durchgriffsdienste auf einem CS-Linux-Server ermöglichen die Kommunikation zwischen einem SNA-Host und lokalen Systemen, die nicht direkt mit dem Host verbunden sind.

Communications Server für Linux bietet TN-Server-Unterstützung für TN3270-, TN3287- und TN3270E-Clients, die unter der Bezeichnung „TN3270-Clients“ zusammengefasst werden. Hinweise zum Konfigurieren dieser Funktion finden Sie im Abschnitt „TN Server konfigurieren“.

Communications Server für Linux unterstützt außerdem TN Redirector für TCP/IP-Hostdurchgriff auf TN3270-, TN3270E-, TN5250- und VT-Clients, die unter der Bezeichnung Telnet-Clients zusammengefasst werden. Hinweise zum Konfigurieren dieser Funktion finden Sie im Abschnitt „TN Redirector konfigurieren“ auf Seite 118.

SNA Gateway stellt Konnektivität zwischen dem Host und den lokalen Systemen zur Verfügung. Sie können LUs auf dem lokalen Knoten für die Unterstützung dieser Funktion konfigurieren (siehe Abschnitt „SNA Gateway konfigurieren“ auf Seite 122) oder eine Schablone für die Unterstützung noch nicht explizit konfigurierter untergeordneter LUs definieren (siehe Abschnitt „DLCs, Ports und Verbindungsnetze definieren“ auf Seite 64).

DLUR unterstützt abhängige LU-Sitzungen zwischen dem Host und Knoten in einem APPN-Netz. Hinweise zum Konfigurieren dieser Funktion finden Sie im Abschnitt „DLUR konfigurieren“ auf Seite 124.

---

### TN Server konfigurieren

TN Server ermöglicht TN3270-Clients, über einen CS-Linux-Transitknoten, der TN Server implementiert, mit einem Host zu kommunizieren. Die TN3270-Clients stellen die Verbindung zum TN-Server über TCP/IP her und benutzen auf dem TN-Server definierte LUs. Die LUs des TN-Servers bauen Sitzungen mit LUs auf dem Host auf, um TN3270-Sitzungen für die Clients zu unterstützen.

Bevor Sie TN Server konfigurieren können, müssen Sie die folgenden Konfigurationsschritte ausführen:

- Definieren Sie den lokalen Knoten wie im Abschnitt „Knoten konfigurieren“ auf Seite 58 beschrieben.
- Konfigurieren Sie einen Port und eine Verbindungsstation für abhängigen Verkehr zwischen dem lokalen Knoten und dem Host. Gehen Sie dazu wie in Kapitel 4, „Konnektivitätskomponenten definieren“, auf Seite 63 beschrieben vor.
- Definieren Sie die TN3270-LUs auf dem lokalen Knoten, die für die Kommunikation mit dem Host verwendet werden. Hinweise zum Hinzufügen der LUs finden Sie im Abschnitt „LUs der Typen 0–3 definieren“ auf Seite 81.
- Wenn Sie LU-Pools verwenden wollen, definieren Sie diese wie im Abschnitt „LU-Pools definieren“ auf Seite 83 beschrieben.

## TN Server konfigurieren

Führen Sie die folgenden Schritte aus, um TN Server zu konfigurieren:

- Konfigurieren Sie für jeden TN3270-Client, der den Server benutzen soll, einen TN-Server-Zugriffseintrag oder einen Standardeintrag, der es jedem Client ermöglicht, auf den Server zuzugreifen (siehe Abschnitt „TN-Server-Zugriffseinträge konfigurieren“).
- Wenn Sie TN3270E- oder TN3287-Clients unterstützen, können Sie für Anzeige- und Drucker-LUs einen Zuordnungseintrag definieren (siehe Abschnitt „TN-Server-Zuordnungseinträge konfigurieren“ auf Seite 117). Auf der Basis dieses Eintrags kann ein TN3270E- oder TN3287-Client einen bestimmten Drucker auswählen (indem er die zugeordnete Anzeige-LU auswählt). Der Client muss berechtigt sein, eine im TN-Server-Zugriffseintrag enthaltene LU auszuwählen.

Über zusätzliche Optionen können Sie Druckerantworten erzwingen, für alle TN3270-Sitzungen eine Keepalive-Methode festlegen und angeben, wie auf den externen LDAP-Server mit der Liste zur Überprüfung der Berechtigung von TN3270-Clients zugegriffen werden soll. Auf diese Optionen können Sie über das Menü **Dienste** im TN-Server-Fenster zugreifen.

## TN-Server-Zugriffseinträge konfigurieren

TN-Server-Zugriffseinträge geben an, welche TN3270-Clients auf den TN-Server zugreifen können und welche LUs sie verwenden sollen. Jeder Zugriffseintrag weist einen TN3270-Client aus, der berechtigt ist, auf den TN-Server zuzugreifen, den TCP/IP-Port, zu dem der Client eine Verbindung herstellt, und die LU bzw. den LU-Pool, die bzw. den der Client verwendet. Sie können auch einen Standardeintrag definieren, der allen TN3270-Clients den Zugriff (mit denselben LUs oder LU-Pools für alle Clients) erlaubt.

TN3270-Clients können den TN-Server nur nutzen, wenn der Knoten, der Port und die Verbindungsstation aktiv sind.

Verwenden Sie für das Konfigurieren eines TN-Server-Zugriffseintrags eine der folgenden Methoden:

### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **TN Server** und im daraufhin erscheinenden Untermenü die Option **TN Server** aus. Wählen Sie im angezeigten Fenster das Teilfenster mit den Clientzugriffsberechtigungen für TN Server und in diesem Teilfenster die Option **Neu** aus.

### Befehlszeilenverwaltungsprogramm

Geben Sie den folgenden Befehl ein:

```
define_tn3270_access
```

**Anmerkung:** Wenn Sie mit dem Befehlszeilenverwaltungsprogramm **snaadmin** oder einer NOF-Anwendung einen TN-Server-Zugriffseintrag definieren, können Sie mit dem Parameter *listen\_local\_address* eine Adresse auf dem lokalen TN-Server-Computer angeben, zu dem der TN3270-Client eine Verbindung herstellen wird. Der Zugriffseintrag wird in diesem Fall nicht im Motif-Verwaltungsprogramm angezeigt, so dass Sie dieses Programm nicht zum Anzeigen oder Verwalten des Eintrags verwenden können. Mit dem Befehlszeilenverwaltungsprogramm oder einer NOF-Anwendung können Sie den Eintrag jedoch verwalten.

## Konfigurationsparameter für TN-Server-Zugriffseintrag

Für die Konfiguration eines TN-Server-Zugriffseintrags sind die folgenden Parameter erforderlich:

### *TN3270-Clientadresse*

Die Adresse, die den TN3270-Client bezeichnet, auf den sich der Zugriffseintrag bezieht.

### **Standardeintrag**

Gestatten des Zugriffs für alle TN3270-Clients.

### **TCP/IP-Name oder -Aliasname**

Gestatten des Zugriffs für einen benannten TN3270-Client. Wenn Sie den TCP/IP-Namen des Clients kennen, wählen Sie diese Option aus, und geben Sie den Namen ein. Auf vielen Computern können Sie den TCP/IP-Namen des Computers mit dem Befehl **hostname** ermitteln.

### **TCP/IP-Adresse**

Gestatten des Zugriffs für eine bestimmte TCP/IP-Adresse. Wenn Sie die TCP/IP-Adresse des TN3270-Clients kennen, wählen Sie diese Option aus, und geben Sie die Adresse ein. Für die Adresse ist eine der folgenden Angaben möglich:

- IPV4-Adresse in Schreibweise mit Trennzeichen (z. B. 193.1.11.100)
- IPV6-Hexadezimaladresse mit Doppelpunkt (z. B. 2001:0db8:0000:0000:0000:0000:1428:57ab oder 2001:db8::1428:57ab)

### *TN3270E unterstützen*

Die vom Knoten bereitgestellte Stufe der TN3270-Unterstützung.

**TN3270** Ausschließliche Unterstützung für das Protokoll TN3270. Bei Auswahl dieser Option wird die Unterstützung für TN3270E-Protokolle inaktiviert, auch wenn diese Protokolle vom Client unterstützt werden.

### **TN3270E**

Unterstützung für TN3270- und TN3270E-Protokolle (die Standardeinstellung).

TN3270- und TN3287-Protokolle werden unabhängig von der ausgewählten Option immer unterstützt.

Für einen AS/400-TN3270-Client muss diese Option auf TN3270E gesetzt werden.

### *TCP/IP-Port-Nummer*

Die TCP/IP-Port-Nummer (auf dem TN-Server) für den Port, zu dem der TN3270-Client eine Verbindung herstellt.

**Anmerkung:** TCP/IP-Ports stehen in keinerlei Beziehung zu SNA-Ports.

Die herkömmliche Port-Nummer für den TN3270-Dienst ist 23. Wenn Sie eine andere Port-Nummer auswählen, die auf dem TN-Server nicht in Gebrauch ist, müssen Sie diese Port-Nummer ebenfalls auf den TN3270-Clients konfigurieren (oder die TN3270-Clients mit einer Option für die Angabe der Port-Nummer starten). Port-Nummern ab 2000 sind mit großer Wahrscheinlichkeit verfügbar. Port-Nummern im Bereich 256–1023 können die Sicherheit geringfügig verbessern, sind jedoch wahrscheinlich bereits vergeben.

## TN Server konfigurieren

Wenn ein TN3270-Client in der Lage sein soll, mehr als eine LU oder mehr als einen LU-Pool zu verwenden, definieren Sie mehrere Zugriffseinträge mit verschiedenen TCP/IP-Port-Nummern, damit Sie die unterschiedlichen LUs bzw. LU-Pools durch Angabe verschiedener Port-Nummern kennzeichnen können.

### *Zugeordnete Anzeige-LU*

Der Name der LU, auf die der TN3270-Client zugreift, wenn sie aktiv ist. Die LU muss eine abhängige LU auf dem lokalen Knoten sein. Sie können statt des Namens einer bestimmten LU auch den Namen eines LU-Pools angeben.

### *Zugeordnete Drucker-LU*

Der Name der Standarddrucker-LU oder des LU-Pools für Clients, die diesen Zugriffseintrag verwenden. Diese LU muss als eine unabhängige LU auf dem lokalen Knoten definiert sein.

### *Zugriff auf bestimmte LU erlauben*

Geben Sie diese Option an, um TN3270E- und TN3287-Clients zu gestatten, für eine Sitzung auf eine bestimmte LU zuzugreifen. (Diese Option ist für TN3270-Clients nicht verfügbar.)

### *Sichere SSL-Sitzung*

Geben Sie diese Option an, wenn diese Sitzung für den Zugriff auf den Server SSL (Secure Sockets Layer) verwenden soll.

Diese Option ist nur verfügbar, wenn Sie die zusätzliche, zur Unterstützung dieser Option erforderliche Software installiert haben. Andernfalls können Sie sie nicht auswählen.

**Anmerkung:** Gibt der Parameter *TCP/IP-Port-Nummer* dieser Sitzung an, dass die Sitzung den TCP/IP-Port des Telnet-Dämons benutzt, sollten Sie kein SSL für diese Sitzung verwenden. Wenn Sie SSL für eine Sitzung verwenden, die den TCP/IP-Port des Telnet-Dämons benutzt, können Telnet-Clients nicht mit **telnet** auf den CS-Linux-Computer zugreifen, solange der Knoten aktiv ist.

### *Clientauthentifizierung ausführen*

Diese Option wird nur angezeigt, wenn Sie die Option *Sichere SSL-Sitzung* ausgewählt haben.

Geben Sie diese Option an, wenn TN Server von der Sitzung die Verwendung der Clientauthentifizierung fordern soll. Der Client muss ein gültiges Zertifikat (Informationen, die den Client als gültigen und zur Verwendung des TN-Servers berechtigten Client ausweisen) senden.

TN Server muss die Gültigkeit des Zertifikats überprüfen und das Zertifikat außerdem mit einer Liste widerrufenen Zertifikate auf einem externen LDAP-Server abgleichen, um sicherzustellen, dass die Berechtigung des Benutzers inzwischen nicht widerrufen wurde. Sie müssen deshalb in der Dialoganzeige mit den erweiterten Parametern für TN Server angeben, wie auf diesen Server zugegriffen werden kann.

### *Sicherheitsstufe*

Gibt die für diese Sitzung erforderliche SSL-Sicherheitsstufe an. Die Sitzung verwendet die höchste Sicherheitsstufe, die sowohl vom Client als auch vom Server unterstützt wird. Sollte der Client keine Unterstützung für die angeforderte oder eine höhere Sicherheitsstufe bieten, wird die Sitzung nicht gestartet.

Diese Option wird nur angezeigt, wenn Sie die Option *Sichere SSL-Sitzung* ausgewählt haben.

Folgende Werte sind gültig:

### **Nur authentifizieren**

Es müssen Zertifikate ausgetauscht werden. Es wird jedoch keine Verschlüsselung verwendet. Diese Option wird in der Regel verwendet, um den für die Verschlüsselung erforderlichen Systemaufwand zu umgehen, wenn der Client eine Verbindung über ein sicheres internes Netz herstellt.

### **Authentifizieren (Minimum)**

Der Client muss vom Server ein Zertifikat anfordern, um dessen Gültigkeit zu überprüfen. Es ist keine Verschlüsselung erforderlich (kann jedoch auf Anforderung des Clients verwendet werden).

### **40-Bit (Minimum)**

Der Client muss mindestens die 40-Bit-Verschlüsselung unterstützen.

### **56-Bit (Minimum)**

Der Client muss mindestens die 56-Bit-Verschlüsselung unterstützen.

### **128-Bit (Minimum)**

Der Client muss mindestens die 128-Bit-Verschlüsselung unterstützen.

### **168-Bit (Minimum)**

Der Client muss mindestens die 168-Bit-Verschlüsselung unterstützen.

**Anmerkung:** Die Verschlüsselung kann nur verwendet werden, wenn Communications Server für Linux mit zusätzlicher Software installiert wurde. Weitere Informationen hierzu finden Sie im Handbuch *Einstieg in Communications Server für Linux*. Unter Umständen können Sie nicht alle aufgelisteten Verschlüsselungsstufen verwenden, weil in Ihrem Land die zur Unterstützung dieser Stufen erforderliche Software nicht erhältlich ist.

## **Zusätzliche Konfiguration**

Führen Sie nach dem Konfigurieren des TN-Server-Zugriffs die folgenden Konfigurationsaufgaben aus:

- Konfigurieren Sie TN-Server-Zugriffseinträge wie im Abschnitt „TN-Server-Zuordnungseinträge konfigurieren“ beschrieben.

## **TN-Server-Zuordnungseinträge konfigurieren**

Ein TN-Server-Zuordnungseintrag definiert eine Zuordnung zwischen einer Drucker-LU und einer Anzeige-LU, aufgrund derer das Protokoll TN3270E oder TN3287 die beiden verbinden kann. Wenn der Zugriffseintrag für den Client die Auswahl einer bestimmten LU gestattet, erlaubt dieser Eintrag dem Client die Auswahl eines bestimmten Druckers durch Angabe der zugeordneten Anzeige-LU.

Verwenden Sie für das Konfigurieren eines TN-Server-Zuordnungseintrags eine der folgenden Methoden:

## TN Server konfigurieren

### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **TN Server** aus. Wählen Sie anschließend im Teilfenster des Fensters für TN-Server-Zuordnungseinträge die Option **Neu** aus.

### Befehlszeilenverwaltungsprogramm

Geben Sie den folgenden Befehl ein:

```
define_tn3270_association
```

### Konfigurationsparameter für TN-Server-Zuordnungseinträge

Für die Konfiguration eines TN-Server-Zuordnungseintrags sind die folgenden Parameter erforderlich:

#### *Anzeige-LU*

Der Name der Anzeige-LU (der auf dem lokalen Knoten definiert sein muss).

#### *Drucker-LU*

Der Name der Drucker-LU (der auf dem lokalen Knoten definiert sein muss). Geben Sie keine Drucker-LU an, die Sie bereits in einem anderen TN-Server-Zuordnungseintrag eingegeben haben.

---

## TN Redirector konfigurieren

TN Redirector ermöglicht TN3270-, TN3270E-, TN5250- und VT-Clients, die unter der Bezeichnung Telnet-Clients zusammengefasst werden, über einen CS-Linux-Transitknoten, der TN Redirector implementiert, mit einem Host zu kommunizieren. Die Verbindung der Clients zum TN-Redirector wird mit TCP/IP hergestellt. Der TN-Redirector baut dann eine separate TCP/IP-Verbindung zum Host auf.

Führen Sie die folgenden Schritte aus, um TN Redirector zu konfigurieren:

- Konfigurieren Sie für jeden Telnet-Client, der den Server benutzen soll, einen TN-Redirector-Zugriffseintrag oder einen Standardeintrag, der es jedem Client ermöglicht, auf den Server zuzugreifen (siehe Abschnitt „TN-Redirector-Zugriffseinträge konfigurieren“).

## TN-Redirector-Zugriffseinträge konfigurieren

TN-Redirector-Zugriffseinträge geben an, welche Telnet-Clients über eine TCP/IP-Verbindung auf den TN-Redirector zugreifen können. Jeder Zugriffseintrag weist einen Telnet-Client aus, der berechtigt ist, auf den TN-Redirector zuzugreifen, den TCP/IP-Port, den der Client für seine Verbindung zu Communications Server für Linux verwendet, den TCP/IP-Port, den Communications Server für Linux für die Verbindung zum Host verwendet, und die SSL-Sicherheitseinstellungen. Sie können auch Standardeinträge definieren, die allen Clients den Zugriff ermöglichen.

Wenn Sie die Absicht haben, allen Clients die Nutzung des TN-Redirectors zu gestatten, und wollen, dass alle Clients dieselbe Hostzugriffskonfiguration verwenden, können Sie einen Standardeintrag definieren.

Telnet-Clients können den TN-Redirector nur verwenden, wenn der Knoten aktiv ist.

Verwenden Sie zum Konfigurieren eines TN-Redirector-Zugriffseintrags eine der folgenden Methoden:

### Motif-Verwaltungsprogramm

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **TN Server** und im daraufhin erscheinenden Untermenü die Option **TN Server** aus. Wählen Sie im angezeigten Fenster das Teilfenster mit den Clientzugriffsberechtigungen für TN Redirector und in diesem Teilfenster die Option **Neu** aus.

### Befehlszeilenverwaltungsprogramm

Geben Sie den folgenden Befehl ein:

```
define_tn_redirect
```

**Anmerkung:** Wenn Sie mit dem Befehlszeilenverwaltungsprogramm **snaadmin** oder einer NOF-Anwendung einen TN-Redirector-Zugriffseintrag definieren, können Sie mit dem Parameter *listen\_local\_address* eine Adresse auf dem lokalen TN-Server-Computer angeben, zu dem der TN3270-Client eine Verbindung herstellen wird. Der Zugriffseintrag wird in diesem Fall nicht im Motif-Verwaltungsprogramm angezeigt, so dass Sie dieses Programm nicht zum Anzeigen oder Verwalten des Eintrags verwenden können. Mit dem Befehlszeilenverwaltungsprogramm oder einer NOF-Anwendung können Sie den Eintrag jedoch verwalten.

### Konfigurationsparameter für TN-Redirector-Zugriffseintrag

Die Konfiguration eines TN-Redirector-Zugriffseintrags umfasst zwei Gruppen von Parametern, eine für den Client und eine für Host-TCP/IP-Verbindungen.

Die Clientparameter lauten wie folgt:

#### *Telnet-Clientadresse*

Die Adresse, die den Telnet-Client bezeichnet, auf den sich der Zugriffseintrag bezieht.

#### **Standardeintrag**

Allen Telnet-Clients den Zugriff erlauben.

#### **TCP/IP-Name oder -Aliasname**

Einem benannten Telnet-Client den Zugriff gestatten. Wenn Sie den TCP/IP-Namen des Clients kennen, wählen Sie diese Option aus, und geben Sie den Namen ein. Auf vielen Computern können Sie den TCP/IP-Namen des Computers mit dem Befehl **hostname** ermitteln.

#### **TCP/IP-Adresse**

Gestatten des Zugriffs für eine bestimmte TCP/IP-Adresse. Wenn Sie die TCP/IP-Adresse des Clients kennen, wählen Sie diese Option aus, und geben Sie die Adresse ein. Für die Adresse ist eine der folgenden Angaben möglich:

- IPv4-Adresse in Schreibweise mit Trennzeichen (z. B. 193.1.11.100)
- IPv6-Hexadezimaladresse mit Doppelpunkt (z. B. 2001:0db8:0000:0000:0000:0000:1428:57ab oder 2001:db8::1428:57ab)

#### *TCP/IP-Port-Nummer*

Die TCP/IP-Port-Nummer (auf dem TN-Server) für den Port, zu dem der Client eine Verbindung herstellt.

**Anmerkung:** TCP/IP-Ports stehen in keinerlei Beziehung zu SNA-Ports.

## TN Redirector konfigurieren

Sie müssen diese Port-Nummer ebenfalls auf den Clients konfigurieren (oder die Clients mit einer Option für die Angabe der Port-Nummer starten). Port-Nummern ab 2000 sind mit großer Wahrscheinlichkeit verfügbar. Port-Nummern im Bereich 256–1023 können die Sicherheit geringfügig verbessern, sind jedoch wahrscheinlich bereits vergeben.

### *Sichere SSL-Sitzung*

Geben Sie diese Option an, wenn diese Sitzung für den Zugriff auf den Server SSL (Secure Sockets Layer) verwenden soll.

Diese Option ist nur verfügbar, wenn Sie die zusätzliche, zur Unterstützung dieser Option erforderliche Software installiert haben. Andernfalls können Sie sie nicht auswählen.

### *Clientauthentifizierung ausführen*

Diese Option wird nur angezeigt, wenn Sie die Option *Sichere SSL-Sitzung* ausgewählt haben.

Geben Sie diese Option an, wenn TN Server von der Sitzung die Verwendung der Clientauthentifizierung fordern soll. Der Client muss ein gültiges Zertifikat (Informationen, die den Client als gültigen und zur Verwendung des TN-Servers berechtigten Client ausweisen) senden.

TN Redirector muss die Gültigkeit des Zertifikats überprüfen und das Zertifikat außerdem mit einer Liste widerrufenen Zertifikate auf einem externen LDAP-Server abgleichen, um sicherzustellen, dass die Berechtigung des Benutzers inzwischen nicht widerrufen wurde. Sie müssen deshalb in der Dialoganzeige mit den erweiterten Parametern für TN Server (die Sie über das Menü **Dienste** des TN-Server-Fensters aufrufen können) angeben, wie auf diesen Server zugegriffen werden kann.

### *Sicherheitsstufe*

Gibt die für die Clientsitzung erforderliche SSL-Sicherheitsstufe an. Die Sitzung verwendet die höchste Sicherheitsstufe, die sowohl vom Client als auch vom Server unterstützt wird. Sollte der Client keine Unterstützung für die angeforderte oder eine höhere Sicherheitsstufe bieten, wird die Sitzung nicht gestartet.

Diese Option wird nur angezeigt, wenn Sie die Option *Sichere SSL-Sitzung* ausgewählt haben.

Folgende Werte sind gültig:

#### **Nur authentifizieren**

Es müssen Zertifikate ausgetauscht werden. Es wird jedoch keine Verschlüsselung verwendet. Diese Option wird in der Regel verwendet, um den für die Verschlüsselung erforderlichen Systemaufwand zu umgehen, wenn der Client eine Verbindung über ein sicheres internes Netz herstellt.

#### **Authentifizieren (Minimum)**

Der Client muss vom Server ein Zertifikat anfordern, um dessen Gültigkeit zu überprüfen. Es ist keine Verschlüsselung erforderlich (kann jedoch auf Anforderung des Clients verwendet werden).

#### **40-Bit (Minimum)**

Der Client muss mindestens die 40-Bit-Verschlüsselung unterstützen.

#### **56-Bit (Minimum)**

Der Client muss mindestens die 56-Bit-Verschlüsselung unterstützen.

### 128-Bit (Minimum)

Der Client muss mindestens die 128-Bit-Verschlüsselung unterstützen.

### 168-Bit (Minimum)

Der Client muss mindestens die 168-Bit-Verschlüsselung unterstützen.

**Anmerkung:** Die Verschlüsselung kann nur verwendet werden, wenn Communications Server für Linux mit zusätzlicher Software installiert wurde. Weitere Informationen hierzu finden Sie im Handbuch *Einstieg in Communications Server für Linux*. Unter Umständen können Sie nicht alle aufgelisteten Verschlüsselungsstufen verwenden, weil in Ihrem Land die zur Unterstützung dieser Stufen erforderliche Software nicht erhältlich ist.

Die Parameter für den Zielhost lauten wie folgt:

#### *Adresse*

Die Adresse, die den Host bezeichnet, auf den sich der Zugriffseintrag bezieht.

#### **TCP/IP-Name oder -Aliasname**

Zugriff auf einen benannten Host. Wenn Sie den TCP/IP-Namen des Hosts kennen, wählen Sie diese Option aus, und geben Sie den Namen ein. Auf vielen Computern können Sie den TCP/IP-Namen des Computers mit dem Befehl **hostname** ermitteln.

#### **TCP/IP-Adresse**

Gestatten des Zugriffs auf eine bestimmte TCP/IP-Adresse. Wenn Sie die TCP/IP-Adresse des Hosts kennen, wählen Sie diese Option aus, und geben Sie die Adresse ein. Für die Adresse ist eine der folgenden Angaben möglich:

- IPV4-Adresse in Schreibweise mit Trennzeichen (z. B. 193.1.11.100)
- IPV6-Hexadezimaladresse mit Doppelpunkt (z. B. 2001:0db8:0000:0000:0000:0000:1428:57ab oder 2001:db8::1428:57ab)

#### *TCP/IP-Port-Nummer*

Die Nummer des TCP/IP-Ports, den TN Redirector für den Zugriff auf den Host verwendet.

**Anmerkung:** TCP/IP-Ports stehen in keinerlei Beziehung zu SNA-Ports.

Sie müssen diese Port-Nummer ebenfalls auf dem Host konfigurieren. Port-Nummern ab 2000 sind mit großer Wahrscheinlichkeit verfügbar. Port-Nummern im Bereich 256–1023 können die Sicherheit geringfügig verbessern, sind jedoch wahrscheinlich bereits vergeben.

#### *Sichere SSL-Sitzung*

Geben Sie diese Option an, wenn TN Redirector für den Zugriff auf den Host SSL (Secure Sockets Layer) verwenden soll.

Diese Option ist nur verfügbar, wenn der Host SSL unterstützt.

#### *Sicherheitsstufe*

Gibt die für die Hostsitzung erforderliche SSL-Sicherheitsstufe an. Die Sitzung verwendet die höchste Sicherheitsstufe, die sowohl vom Host als

## TN Redirector konfigurieren

auch vom Server unterstützt wird. Sollte der Host keine Unterstützung für die angeforderte oder eine höhere Sicherheitsstufe bieten, wird die Sitzung nicht gestartet.

Diese Option wird nur angezeigt, wenn Sie die Option *Sichere SSL-Sitzung* ausgewählt haben.

Folgende Werte sind gültig:

### **Nur authentifizieren**

Es müssen Zertifikate ausgetauscht werden. Es wird jedoch keine Verschlüsselung verwendet. Diese Option wird in der Regel verwendet, um den für die Verschlüsselung erforderlichen Systemaufwand zu umgehen, wenn die Hostverbindung über ein sicheres internes Netz hergestellt wird.

### **Authentifizieren (Minimum)**

Der Host muss vom Server ein Zertifikat anfordern, um dessen Gültigkeit zu überprüfen. Es ist keine Verschlüsselung erforderlich (kann jedoch auf Anforderung des Hosts verwendet werden).

### **40-Bit (Minimum)**

Der Host muss mindestens die 40-Bit-Verschlüsselung unterstützen.

### **56-Bit (Minimum)**

Der Host muss mindestens die 56-Bit-Verschlüsselung unterstützen.

### **128-Bit (Minimum)**

Der Host muss mindestens die 128-Bit-Verschlüsselung unterstützen.

### **168-Bit (Minimum)**

Der Host muss mindestens die 168-Bit-Verschlüsselung unterstützen.

**Anmerkung:** Die Verschlüsselung kann nur verwendet werden, wenn Communications Server für Linux mit zusätzlicher Software installiert wurde. Weitere Informationen hierzu finden Sie im Handbuch *Einstieg in Communications Server für Linux*. Unter Umständen können Sie nicht alle aufgelisteten Verschlüsselungsstufen verwenden, weil in Ihrem Land die zur Unterstützung dieser Stufen erforderliche Software nicht erhältlich ist.

---

## SNA Gateway konfigurieren

Normalerweise erfordert eine abhängige LU-Sitzung eine direkte DFV-Verbindung zum Host. Ein Knoten, der Communications Server für Linux ausführt und eine direkte DFV-Verbindung zum Host hat, kann aber auch SNA-Gateway-Funktionen für LUs auf untergeordneten Computern bereitstellen, so dass diese über die DFV-Verbindung des CS-Linux-Knotens auf den Host zugreifen können. Der untergeordnete Computer muss über eine SNA-PU des Typs 2.0 oder 2.1 verfügen, um die abhängige Kommunikation mit dem Host unterstützen zu können. Der untergeordnete Computer könnte beispielsweise ein weiterer Computer mit Communications Server für Linux sein.

Bei Verwendung des Features SNA Gateway werden alle Daten, die zwischen dem Host und dem untergeordneten Computer übertragen werden, über den lokalen CS-Linux-Knoten geleitet. Dadurch benötigt der untergeordnete Computer keine Direktverbindung und kann die Hostverbindung gemeinsam mit dem CS-Linux-

Computer oder mit anderen untergeordneten Computern nutzen. Sie können z. B. mehrere untergeordnete Computer installieren, die über ein lokales Token-Ring-Netz mit dem CS-Linux-Computer verbunden sind. In diesem Fall greifen alle Computer auf dieselbe SDLC-Standfernleitung des CS-Linux-Computers zum Host zu.

Die Verwendung von SNA Gateway vereinfacht darüber hinaus die Konfiguration auf dem Host. Die Hostkonfiguration muss nur den CS-Linux-Computer und dessen Hostübertragungsleitung umfassen. Die LUs auf den untergeordneten Computern werden zusammen mit den Ressourcen des CS-Linux-Computers konfiguriert. Der Host hat keine Kenntnis von der Verwendung der Funktion SNA Gateway.

Bevor Sie SNA Gateway konfigurieren, müssen Sie die folgenden Konfigurationsschritte ausführen:

- Definieren Sie den lokalen Knoten wie im Abschnitt „Knoten konfigurieren“ auf Seite 58 beschrieben.
- Konfigurieren Sie einen Port und eine Verbindungsstation für abhängigen Verkehr zwischen dem lokalen Knoten und dem Host. Gehen Sie dazu wie in Kapitel 4, „Konnektivitätskomponenten definieren“, auf Seite 63 beschrieben vor. Konfigurieren Sie außerdem Ports und Verbindungsstationen für abhängigen Verkehr zwischen dem lokalen Knoten und den untergeordneten Knoten. Bei untergeordneten Verbindungen können Sie für den Port eine Schablone für die Unterstützung impliziter untergeordneter LUs (LUs, die nicht explizit auf dem lokalen Knoten definiert sind) konfigurieren.
- Definieren Sie auf dem lokalen Knoten die LUs, die für die Kommunikation mit dem Host (den übergeordneten LUs) verwendet werden. Übergeordnete LUs, einschließlich abhängiger LUs des Typs 6.2, müssen im Konfigurationsdialog für LUs der Typen 0–3 unter Angabe eines unbeschränkten LU-Typs definiert werden. Hinweise zum Hinzufügen der LUs finden Sie im Abschnitt „LUs der Typen 0–3 definieren“ auf Seite 81.
- Wenn Sie LU-Pools verwenden wollen, definieren Sie diese wie im Abschnitt „LU-Pools definieren“ auf Seite 83 beschrieben.

Zum Aktivieren von SNA Gateway müssen Sie LUs auf dem lokalen Knoten für die Unterstützung von Sitzungen mit untergeordneten Workstations konfigurieren. (Wenn Sie für den Port eine Schablone für die Unterstützung impliziter untergeordneter LUs konfiguriert haben, müssen Sie untergeordnete LUs möglicherweise nicht explizit definieren.) Die auf dem lokalen Knoten definierten LUs werden als „untergeordnete LUs“ bezeichnet. Zum Konfigurieren untergeordneter LUs benötigen Sie die auf den untergeordneten Knoten verwendeten LU-Nummern und den Namen der Host-LU. (Die auf den untergeordneten Knoten definierten LUs können LUs eines beliebigen abhängigen LU-Typs sein.)

Verwenden Sie für das Konfigurieren untergeordneter LUs eine der folgenden Methoden:

### **Motif-Verwaltungsprogramm**

Wählen Sie im Menü **Dienste** des Knotenfensters den Eintrag **SNA Gateway** und dann die Option **Neue untergeordnete LU** aus.

### **Befehlszeilenverwaltungsprogramm**

Setzen Sie einen der folgenden Befehle ab:

```
define_downstream_lu
```

```
define_downstream_lu_range
```

### Konfigurationsparameter für untergeordnete LUs

Für die Konfiguration einer untergeordneten LU sind die folgenden Parameter erforderlich:

#### *Name der untergeordneten LU*

Ein Name für jede untergeordnete LU. Der LU-Name wird nur für die lokale Identifizierung der LU verwendet und muss nicht mit einem auf dem übergeordneten Knoten konfigurierten Wert übereinstimmen.

Wenn Sie einen LU-Bereich definieren, geben Sie einen aus 1–5 Zeichen bestehenden Basisnamen an. Communications Server für Linux fügt zum Basisnamen eine aus drei Dezimalzahlen bestehende Zeichenfolge hinzu, um für jede von Ihnen angegebene LU-Nummer einen LU-Namen zu erstellen.

#### *Name der untergeordneten PU*

Der Name der Verbindungsstation zum untergeordneten Knoten.

#### *LU-Nummer*

Die LU-Nummer muss mit der auf dem untergeordneten Knoten definierten LU-Nummer übereinstimmen. Sollten Sie nicht wissen, welche LU-Nummer Sie verwenden müssen, wenden Sie sich an den zuständigen Planungsbeauftragten für das SNA-Netz.

Sie können mehrere LUs mit fortlaufenden LU-Nummern konfigurieren, indem Sie einen LU-Bereich definieren.

#### *Name der übergeordneten LU*

Der Name der Host-LU oder eines LU-Pools, mit der bzw. dem die untergeordneten LUs kommunizieren sollen.

#### *Verzögerte Anmeldung*

Um die Systemstartzeit zu verkürzen, ruft Communications Server für Linux eine Anmeldeanzeige auf, ohne eine übergeordnete LU zuzuordnen. Ein 3270-Benutzer muss eine Taste drücken, bevor ihm eine übergeordnete LU zugeordnet wird.

#### *Zeitlimit zulassen*

Um die Anzahl der erforderlichen LUs zu reduzieren, wird nach Ablauf dieser Zeit für LUs ohne aktive PLU-SLU-Sitzung die Zuordnung zur übergeordneten LU aufgehoben.

### Zusätzliche Konfiguration

Führen Sie nach dem Konfigurieren der untergeordneten LUs für SNA Gateway die folgenden Konfigurationsaufgaben aus:

- Wie Sie Benutzeranwendungen konfigurieren, können Sie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 nachlesen.

---

## DLUR konfigurieren

Normalerweise erfordert eine abhängige LU-Sitzung eine direkte DFV-Verbindung zum Host. Wenn in einem APPN-Netz viele Knoten (einschließlich eines Hostknotens) verbunden sind, haben einige von ihnen unter Umständen anstelle einer Direktverbindung zum Host eine indirekte Verbindung über einen anderen Knoten. LUs auf diesen indirekt mit dem Host verbundenen Knoten können ohne eine Direktverbindung keine abhängigen LU-Sitzungen mit dem Host aufbauen.

Dependent LU Requester (DLUR) ist ein APPN-Feature, das diese Einschränkung aufhebt. DLUR kann auf einem APPN Knoten (z. B. einem CS-Linux-Knoten) konfiguriert werden. Die DLUR-Funktion arbeitet mit DLUS (Dependent LU Server) auf dem Host zusammen, um Sitzungen von abhängigen LUs auf dem DLUR-Knoten über das APPN-Netz zum DLUS-Host weiterzuleiten.

Die Route zum Host kann sich über mehrere Knoten erstrecken und von den APPN-Funktionen für Netzverwaltung, für dynamische Ressourcenlokalisierung und Routenberechnung profitieren. DLUR muss auf dem Knoten verfügbar sein, auf dem die LUs definiert sind. DLUS muss auf dem Hostknoten verfügbar sein. Auf den Transitknoten der Sitzungsrouten müssen Sie die DLUR-Funktion nicht aktivieren.

**Anmerkung:** Auf einem LEN-Knoten können Sie DLUR nicht konfigurieren.

Wenn der CS-Linux-DLUR-Knoten ein Netzknoten ist, kann er auch Durchgriff-DLUR-Funktionen für abhängige LUs auf untergeordneten Computern, die mit dem CS-Linux-Knoten verbunden sind, bereitstellen. (Diese Funktion wird nur von Netzknoten unterstützt.) Diese untergeordneten LUs können ebenso wie knoteninterne LUs DLUR auf dem CS-Linux-Knoten nutzen, um über das Netz auf den Host zuzugreifen.

**Anmerkung:** Auf einem Endknoten können Sie Durchgriff-DLUR nicht konfigurieren.

Die für die DLUR-Konfiguration erforderlichen Schritte richten sich danach, ob sich die abhängigen LUs auf dem lokalen Knoten oder auf untergeordneten Knoten befinden.

Zum Konfigurieren der DLUR-Unterstützung auf dem lokalen Knoten müssen Sie die folgenden Konfigurationsschritte ausführen:

1. Definieren Sie den lokalen Knoten wie im Abschnitt „Knoten konfigurieren“ auf Seite 58 beschrieben. Wenn Sie Durchgriff-DLUR-Unterstützung für untergeordnete Knoten bereitstellen wollen, definieren Sie den Knoten als APPN-Netzknoten.
2. Konfigurieren Sie die Konnektivität zum APPN-Netz. Für die APPN-Konnektivität sind mindestens ein Port und eine Verbindungsstation für unabhängigen Verkehr zwischen dem lokalen Knoten und dem benachbarten APPN-Netzknoten erforderlich. Lesen Sie die diesbezüglichen Informationen im Abschnitt Kapitel 4, „Konnektivitätskomponenten definieren“, auf Seite 63.
3. Definieren Sie eine DLUR-PU auf dem lokalen Knoten wie im Abschnitt „DLUR-PUs definieren“ auf Seite 77 beschrieben. (Die DLUR-PU unterstützt Konnektivität zum Host.)
4. Wenn Sie DLUR für die Unterstützung von LUs auf dem lokalen Knoten konfigurieren wollen, müssen Sie die LUs auf dem lokalen Knoten hinzufügen. Gehen Sie dazu wie in Kapitel 7, „Benutzeranwendungen konfigurieren“, auf Seite 111 beschrieben vor. Die LUs können für die Unterstützung von 3270-Terminals, 3270-Druckern oder LUA konfiguriert werden. In Abhängigkeit von den Anforderungen der von den LUs unterstützten Benutzeranwendungen können weitere Konfigurationsschritte erforderlich sein.

## DLUR konfigurieren

Zum Konfigurieren der Durchgriff-DLUR-Unterstützung für untergeordnete Knoten müssen Sie die folgenden Konfigurationsschritte ausführen:

1. Definieren Sie den lokalen Knoten als APPN-Netz-knoten (siehe Abschnitt „Knoten konfigurieren“ auf Seite 58).
2. Konfigurieren Sie die Konnektivität zu den untergeordneten Knoten. Konfigurieren Sie Ports und Verbindungsstationen für abhängigen Verkehr zwischen dem lokalen Knoten und allen untergeordneten Knoten. Gehen Sie dazu wie in Kapitel 4, „Konnektivitätskomponenten definieren“, auf Seite 63 beschrieben vor. (Sie müssen keine DLUR-PU für die DLUR-Unterstützung für untergeordnete Knoten definieren.)
3. Ein untergeordneter Knoten kann mehrere PUs unterstützen. In einem solchen Fall ist jede untergeordnete PU einer anderen Verbindung zugeordnet, so dass Sie zwischen dem CS-Linux-DLUR-Knoten und dem untergeordneten Knoten mehrere Verbindungen konfigurieren und für jede Verbindung den Namen der untergeordneten PU kennen müssen.

---

## Kapitel 9. Communications Server für Linux mit NetView verwalten

Im Lieferumfang von Communications Server für Linux ist eine Funktion für ferne Befehle (RCF, Remote Command Facility) enthalten, die mit dem Programm NetView auf einem Host zusammenarbeitet und dem NetView-Bediener gestattet, mit dem Hostprogramm NetView Befehle für den CS-Linux-Computer abzusetzen. (Eine Kurzübersicht über die NetView- und RCF-Befehle finden Sie im Abschnitt „Hostprogramm NetView verwenden“.)

Die RCF-Einrichtung von Communications Server für Linux stellt die beiden folgenden Funktionen bereit:

- Mit der Servicepunkt-Befehlsfunktion (SPCF, Service Point Command Facility) kann ein NetView-Bediener über NetView CS-Linux-Verwaltungsbefehle absetzen. Die Syntax dieser Befehle ist mit der für das Befehlszeilenverwaltungsprogramm **snaadmin** identisch. Eine Beschreibung dieser Funktion finden Sie im Abschnitt „SPCF verwenden“ auf Seite 130.
- Mit der UNIX-Befehlsfunktion (UCF, UNIX Command Facility) kann ein NetView-Bediener über NetView Befehle des Betriebssystems Linux absetzen. Eine Beschreibung dieser Funktion finden Sie im Abschnitt „UCF verwenden“ auf Seite 131.

Auf beide genannten Funktionen kann über eine NetView-Konsole zugegriffen werden. Die Syntax der abgesetzten Befehle ist für beide Funktionen identisch.

---

### Hostprogramm NetView verwenden

Die RCF-Einrichtung von Communications Server für Linux arbeitet mit dem Programm NetView auf einem Host zusammen. Auf dem Host muss NetView ab Version 1 Release 2 installiert sein. Communications Server für Linux bietet keine Unterstützung für NetView Version 1 Release 1.

Zur Verwendung des Programms NetView benötigen Sie Folgendes:

- Anmelde-ID und Kennwort für das Hostprogramm NetView (diese Informationen kann Ihnen das für den Host zuständige Personal geben)
- Servicepunktname für Communications Server für Linux, der auf dem Host für das Programm NetView definiert ist (diese Information kann Ihnen das für den Host zuständige Personal geben)
- DLC, Port und Verbindungsstation für den Zugriff auf den Host, auf dem das Programm NetView ausgeführt wird.

Sie können die Funktion RCF testen, indem Sie statt über eine Direktverbindung mit der 3270-Emulation von Communications Server für Linux auf NetView zugreifen. Für einen solchen Test benötigen Sie Folgendes:

- Eine auf dem Host konfigurierte 3270-LU
- Eine 3270-Sitzung, die diese LU benutzt.

Wenden Sie sich wegen der erforderlichen Konfigurationsdaten an den Administrator des Hosts.

## Hostprogramm NetView verwenden

Führen Sie die folgenden Schritte aus, um auf das Programm NetView zuzugreifen:

1. Stellen Sie sicher, dass Communications Server für Linux mit einer Knotenkonfigurationsdatei gestartet wurde, die eine Definition der RCF-Zugriffsparameter (den Eintrag `define_rcf_access`) enthält.
2. Wenn Sie mit der 3270-Emulation auf das Programm NetView zugreifen wollen, starten Sie das 3270-Emulationsprogramm und aktivieren Sie die Sitzung mit dem Host.
3. Führen Sie die Anweisungen des Hostadministrators für das Starten von NetView und das Anmelden aus. (Die Operationsfolge kann je nach NetView-Version verschieden sein.)
4. Setzen Sie die erforderlichen SPCF- oder UCF-Befehle ab.
5. Wenn Sie mit der 3270-Emulation auf NetView zugreifen, beenden Sie die 3270-Emulation nach dem Absetzen der Befehle gemäß den Anweisungen in der 3270-Dokumentation.

## NetView-Bildschirmanzeige

Das Aussehen der NetView-Anzeige ist je nach NetView-Version und Host verschieden. Eine typische Anzeige ist in Abb. 19 dargestellt.

Am unteren Rand der Anzeige befindet sich ein Eingabebereich. In diesem Bereich können Sie Befehle eingeben. Die Zeile `???` unterteilt die Hauptanzeige (in der die Antworten von NetView auf Ihre Befehle angezeigt werden) vom Eingabebereich.

```
NCCF          N E T V I E W          [SCAN DDAC12  07/18/95  13:52:24 A
              RUNCMD SP=ADCDPU01,APL=NODE,START_DLC,DLC_NAME=TOKR01
              COMMAND ISSUED SUCCESSFULLY

???
```

```
runcmd sp=abcdpu01,appl=node,query_node
```

Abbildung 19. Beispiel für eine NetView-Anzeige

## Größe des Eingabebereichs ändern

Der Eingabebereich besteht standardmäßig aus nur einer Zeile. Für einige längere Befehle benötigen Sie jedoch mehr als eine Zeile. Bei einigen Versionen von NetView können Sie mit dem Befehl `input` einen Eingabebereich von einer, zwei oder drei Zeilen festlegen. Geben Sie dazu den folgenden Befehl ein:

`input n`

In diesem Befehl hat `n` den Wert 1, 2 oder 3 und gibt die Zahl der von Ihnen gewünschten Zeilen an. Sollte dieser Befehl in der NetView-Version, mit der Sie arbeiten, nicht ausgeführt werden können, wenden Sie sich an die zuständige NetView-Benutzerunterstützung.

## Übersicht über die Syntax der RCF-Befehle

Die RCF-Befehlssyntax gilt sowohl für SPCF- als auch für UCF-Befehle.

`runcmd sp=Servicepunktname, appl=Komponente, Befehlstext`

NetView sendet mit dem Dienstprogramm **runcmd** eine Befehlsfolge an das ferne System. Der Befehl enthält die folgenden Parameter:

*sp=Servicepunktname*

Geben Sie den (in NetView definierten) Namen des Servicepunkts an, der dem CS-Linux-Knoten entspricht. Diese Information kann Ihnen das für den NetView-Host zuständige Personal geben.

*appl=Komponente*

Geben Sie wie folgt den Namen der CS-Linux-Komponente an, an die NetView den Befehl senden soll:

**node** Der dem Servicepunktnamen *Servicepunktname* zugeordnete CS-Linux-Knoten (für SPCF-Befehle)

**unix** Das auf dem CS-Linux-Computer ausgeführte UCF-Dämonprogramm, das dem Servicepunktnamen *Servicepunktname* zugeordnet ist (für UCF-Befehle)

*Befehlstext*

Gibt den Text des abgesetzten Befehls an. Für SPCF ist dies ein an das Befehlszeilenverwaltungsprogramm von Communications Server für Linux abgesetzter Befehl. Für UCF ist es ein an das Betriebssystem Linux abgesetzter Befehl. Weitere Informationen zu Befehlen, die Sie verwenden können, finden Sie im Abschnitt „Einschränkungen für über SPCF abgesetzte Verwaltungsbefehle“ auf Seite 130 bzw. im Abschnitt „Zulässige Befehle“ auf Seite 132.

## Großbuchstaben und Escape-Zeichen

Im Gegensatz zu Linux unterscheidet das Programm NetView nicht zwingend zwischen alphabetischen Zeichen in Groß- oder Kleinschreibung. Der NetView-Befehl **netvasis** kann verwendet werden, um Eingaben in gemischter Groß-/Kleinschreibung für **runcmd** bereitzustellen. Die RCF-Funktion von Communications Server für Linux kann jedoch nicht feststellen, ob **netvasis** verwendet wird. Da RCF nicht feststellen kann, ob ein vom Host empfangenes alphabetisches Zeichen ursprünglich groß oder klein geschrieben war, geht RCF davon aus, dass die Zeichen in Kleinschreibung empfangen werden. Außerdem unterstützt möglicherweise der Hostzeichensatz keine eckigen Klammern [ und ], die für einige Befehle erforderlich sind.

RCF bietet mit Hilfe des umgekehrten Schrägstrichs \ wie folgt Unterstützung für Großbuchstaben und eckige Klammern:

- Wenn Sie einen Großbuchstaben in die Befehlsfolge aufnehmen wollen, müssen Sie vor diesen Buchstaben einen umgekehrten Schrägstrich setzen. Alle alphabetischen Zeichen, denen kein umgekehrter Schrägstrich vorangestellt ist, werden als Kleinbuchstaben interpretiert.
- Wenn Sie die eckigen Klammern [ und ] in die Zeichenfolge aufnehmen wollen, verwenden Sie die Zeichenfolgen \ ( und \ ).
- Sollten Sie den umgekehrten Schrägstrich \ selbst in die Zeichenfolge aufnehmen wollen, geben Sie ihn zweimal ein.

Folgt auf einen einzelnen umgekehrten Schrägstrich ein anderes, nicht alphabetisches Zeichen, wird der umgekehrte Schrägstrich ignoriert und das Zeichen wird nicht verändert.

## Hostprogramm NetView verwenden

In Tabelle 3 finden Sie einige Beispiele.

Tabelle 3. Verwendung von Escape-Zeichen in RCF-Befehlen

Zu produzierende Zeichen	Eingabe
ABcd	\a\bcd
[ ]	\( \)
\a	\\a
\[	\\\[

Die Escape-Zeichen, die Sie normalerweise in der Linux-Befehlszeile verwenden, damit die Linux-Shell keine Sonderzeichen interpretiert, sind für RCF nicht erforderlich. Verwenden Sie beispielsweise keine Escape-Zeichen in Zeichenfolgen, die die Zeichen \* oder \$ enthalten, wie Sie das bei Eingabe dieser Zeichen in der Linux-Befehlszeile tun würden. Beachten Sie beim Absetzen von Verwaltungsbefehlen mit SPCF, dass bei der Schreibung von Konstantennamen wie LIST\_FROM\_NEXT die Groß-/Kleinschreibung keine Rolle spielt. Sie müssen keine Escape-Zeichen eingeben, um aus den Zeichen Großbuchstaben zu machen.

---

## SPCF verwenden

Mit SPCF können Sie Befehle von der NetView-Konsole aus absetzen, um das aktive CS-Linux-System zu verwalten. Diese Befehle stimmen mit denen überein, die Sie im CS-Linux-Befehlszeilenverwaltungsprogramm **snaadmin** absetzen können. (Informationen hierzu finden Sie in der Veröffentlichung *Communications Server for Linux Administration Command Reference*.)

Informationen zur Syntax eines SPCF-Befehls finden Sie im Abschnitt „Übersicht über die Syntax der RCF-Befehle“ auf Seite 128. Der Befehltext, der auf den Parameter *appl=node* folgt, ist ein an das CS-Linux-Befehlszeilenverwaltungsprogramm abgesetzter Befehl, der dasselbe Format hat wie beim Absetzen über das Programm **snaadmin** in der Linux-Befehlszeile. Informationen zur Syntax von Verwaltungsbefehlen und zu den Parametern einzelner Befehle können Sie der Veröffentlichung *Communications Server for Linux Administration Command Reference* entnehmen.

## Einschränkungen für über SPCF abgesetzte Verwaltungsbefehle

Sie können nicht die Befehlszeilenoption **-i** verwenden, um Eingaben aus einer Datei oder von der Standardeingabe anzugeben. Alle Befehle müssen direkt auf der NetView-Konsole eingegeben werden.

Die Befehle **query\_\*** können Sie wie bei Eingabe in der Linux-Befehlszeile mit den Befehlszeilenoptionen **-a** (alle Einträge zurückgeben) und **-d** (ausführliche Informationen zurückgeben) verwenden.

Im Interesse einer höheren Sicherheit können Sie Communications Server für Linux so konfigurieren, dass mit SPCF nur bestimmte Arten von Befehlen abgesetzt werden können. Sie können beispielsweise fernem Benutzern gestatten, die Befehle **query\_\*** abzusetzen, und sie daran hindern, CS-Linux-Komponenten zu aktivieren oder zu inaktivieren. Für die folgenden Befehlsgruppen können Sie den Zugriff separat steuern:

- Die Befehle **define\_\***, **set\_\***, **delete\_\***, **add\_\*** und **remove\_\*** sowie **init\_node**
- Die Befehle **query\_\***

- Die Aktionsbefehle **start\_\***, **stop\_\***, **activate\_\*** und **deactivate\_\***, aber auch **aping**, **initialize\_session\_limit**, **change\_session\_limit** und **reset\_session\_limit**.

Weitere Informationen zum Konfigurieren von Sicherheitsoptionen für SPCF können Sie der Beschreibung zum Befehl **define\_rcf\_access** in der Veröffentlichung *Communications Server for Linux Administration Command Reference* entnehmen.

## Beispiele für SPCF-Befehle

Das folgende Beispiel zeigt, wie Sie mit SPCF den Befehl **define\_lu\_0\_to\_3** absetzen können. In diesem Beispiel wird für die Angabe der Großbuchstaben in den beiden Zeichenfolgen LU\$01 und PU2 der umgekehrte Schrägstrich verwendet. Die Zeichen des Konstantennamens `3270_display_model_2` müssen nicht groß geschrieben werden, weil das Programm **snaadmin** diese Zeichenfolge auch in Kleinbuchstaben akzeptiert.

```
runcmd sp=Mein_Servicepunktname, appl=node, define_lu_0_to_3,
      lu_name=\\u$01,
      nau_address=1, pu_name=\\p\u2, lu_model=3270_display_model_2
```

Das folgende Beispiel zeigt, wie Sie mit SPCF den Befehl **query\_lu\_0\_to\_3** absetzen können. Die Option **-a** gibt an, dass alle Einträge zurückgegeben werden sollen, weshalb kein LU-Name oder PU-Name angegeben werden muss. Die Option **-d** gibt an, dass ausführliche Informationen zurückgegeben werden sollen, weshalb dies nicht mit dem Parameter *list\_options* angegeben werden muss. Diese beiden Optionen führen zum selben Ergebnis wie im Programm **snaadmin**.

```
runcmd sp=Mein_Servicepunktname, appl=node, -a -d query_lu_0_to_3
```

---

## UCF verwenden

Mit UCF kann ein NetView-Bediener Linux-Befehle auf einem CS-Linux-Computer absetzen, indem er den Befehlstext auf der NetView-Konsole eingibt. Darüber hinaus kann er auch die Ausgabe dieser Befehle anzeigen. Diese Funktion ist nicht auf Befehle für Communications Server für Linux beschränkt. Mit Ausnahme der im Abschnitt „Zulässige Befehle“ auf Seite 132 genannten Befehle können alle Arten von Befehlen abgesetzt werden.

Über UCF kann ein ferner Bediener die Aktivitäten auf dem CS-Linux-Computer überwachen, Fehler bestimmen und in einigen Fällen sogar Korrekturmaßnahmen ergreifen.

Sie können angeben, ob Communications Server für Linux UCF über den Befehl **define\_rcf\_access** unterstützen soll (lesen Sie die diesbezüglichen Informationen in der Veröffentlichung *Communications Server for Linux Administration Command Reference*). Ist in der Konfiguration angegeben, dass UCF unterstützt wird, startet Communications Server für Linux das UCF-Dämonprogramm, wenn der Knoten gestartet wird. Der UCF-Dämon verarbeitet Linux-Befehle von UCF, indem er für jeden Befehl eine neue Linux-Shell startet und den Befehl in dieser Shell ausführt. Enthält die Konfiguration keine Angabe zur UCF-Unterstützung, startet Communications Server für Linux dieses Programm nicht.

Die Konfiguration gibt den Namen des UCF-Benutzers an, der mit einem gültigen Anmeldenamen auf dem CS-Linux-Computer übereinstimmen muss. Die UCF-Shell wird unter Verwendung des Shell-Programms, der Anmelde-ID, der Berechtigungen und der Datei **.login** oder **.profile**, die für diesen Benutzer angegeben sind,

gestartet. (Wenn kein Shell-Programm angegeben wurde, wird `/bin/sh` verwendet.) Dies bedeutet, dass die normalen Linux-Sicherheitsfeatures zur Einschränkung des Zugriffs von UCF-Benutzern auf Dateien und Befehle genutzt werden können und so der Bereich der verfügbaren UCF-Befehle eingeschränkt werden kann.

Weitere Informationen zur UCF-Konfiguration können Sie der Beschreibung zum Befehl `define_rcf_access` in der Veröffentlichung *Communications Server for Linux Administration Command Reference* entnehmen.

## Syntax der UCF-Befehle

UCF-Befehle haben folgende Syntax:

**runcmd** *sp=Servicepunktname*, **appl=unix**, *UNIX-Befehl*

NetView sendet mit dem Dienstprogramm **runcmd** einen Befehl an das ferne System. Der Befehl enthält die folgenden Parameter:

*sp=Servicepunktname*

Geben Sie für *Servicepunktname* den Namen des Servicepunkts an, wie er in NetView definiert ist. Diese Information kann Ihnen das für den NetView-Host zuständige Personal geben.

**appl=unix**

Mit dieser Angabe weisen Sie NetView an, den Befehl an das UCF-Dämonprogramm auf dem CS-Linux-Computer zu senden, der dem Servicepunktnamen *Servicepunktname* zugeordnet ist.

*UNIX-Befehl*

Geben Sie den Linux-Betriebssystembefehl ein. Diesen Befehl müssen Sie mit Ausnahme der Escape-Zeichen für Großbuchstaben oder eckige Klammern wie in der Linux-Befehlszeile eingeben. (Lesen Sie die diesbezüglichen Informationen im Abschnitt „Übersicht über die Syntax der RCF-Befehle“ auf Seite 128.)

Die Escape-Zeichen, die Sie normalerweise in der Linux-Befehlszeile verwenden, damit die Linux-Shell keine Sonderzeichen interpretiert, sind für UCF nicht erforderlich. Verwenden Sie beispielsweise keine Escape-Zeichen in Zeichenfolgen, die die Zeichen `*` oder `$` enthalten, wie Sie das bei Eingabe dieser Zeichen in der Linux-Befehlszeile tun würden.

## Zulässige Befehle

UCF ist für Befehle ausgelegt, die (unabhängig davon, ob eine Ausgabe produziert wird oder nicht) vollständig ohne weiteres Eingreifen des Benutzers ausgeführt werden können. Sie können beispielsweise den Befehl `cat Dateiname` absetzen, der nach Anzeige des Inhalts von *Dateiname* vollständig ausgeführt wird, oder den Befehl `mv Dateiname1 Dateiname2`, der ohne Generieren einer Ausgabe vollständig ausgeführt wird, sofern kein Fehler auftritt.

Die von einem UCF-Befehl generierte Ausgabe wird nach Abschluss des Linux-Betriebssystembefehls an UCF zurückgegeben. Daraus ergeben sich die folgenden Einschränkungen:

- Alle nach Abschluss des Befehls generierten Ausgaben werden nicht an UCF zurückgegeben. Wenn Sie einen Befehl beispielsweise mit dem Zeichen `&` absetzen, um ihn im Hintergrund auszuführen, empfängt UCF die Betriebssystemnachricht mit der Prozess-ID des Hintergrundbefehls, jedoch keine der nachfolgend generierten Ausgaben. Ebenso können Sie mit UCF einen Dämonprozess starten, aber keine vom Prozess generierte Ausgabe anzeigen.

- UCF kann nicht für einen Befehl verwendet werden, der weitere Eingaben des Benutzers erfordert, bevor die Ausführung des Befehls abgeschlossen ist (z. B. der Befehl `vi Dateiname`, der einen interaktiven Prozess startet, oder ein Befehl wie `tail -f Dateiname`, der solange ausgeführt wird, bis der Benutzer ihn stoppt).

Da alle Linux-Befehle mit der Anmelde-ID und den Berechtigungen des konfigurierten UCF-Benutzers ausgeführt werden, sind die gültigen Befehle durch die Zugriffsrechte des UCF-Benutzers eingeschränkt. Insbesondere sind keine root- oder superuser-Befehle erlaubt. Weitere Informationen hierzu finden Sie im Abschnitt „UCF-Sicherheit“ auf Seite 134.

## Beispiel für einen UCF-Befehl

Nachstehend sehen Sie ein Beispiel für einen UCF-Befehl, wie Sie ihn in NetView eingeben würden:

```
runcmd sp=Mein_Servicepunktname, appl=unix, grep \temp \((ab\)*.c >
\t\e\m\p.out
```

Der auf dem Linux-Computer ausgeführte Befehl würde wie folgt aussehen:

```
grep Temp [ab]*.c >TEMP.out
```

## Ausgabe von Linux-Systembefehlen

Wenn ein Befehl erfolgreich abgesetzt wurde, erscheinen in der NetView-Anzeige die folgenden Nachrichten:

```
= = = EXECUTING UNIX COMMAND = = =
(Alle Ausgaben des Befehls einschließlich Fehlermeldungen)
= = = UNIX COMMAND COMPLETED = = =
```

Diese Nachrichten müssen nicht alle gleichzeitig in der NetView-Anzeige erscheinen. Die Nachricht **EXECUTING UNIX COMMAND** wird angezeigt, sobald das UCF-Dämonprogramm den Befehl empfangen und die Steuerung wieder an den NetView-Bediener übergeben hat. Alle Ausgaben des Befehls werden an NetView gesendet, sobald sie generiert wurden, und können in der Anzeige als Reihe gesonderter Nachrichten erscheinen. Die Nachricht **UNIX COMMAND COMPLETED** wird angezeigt, wenn der Linux-Befehl abgeschlossen und die zugehörige Shell beendet ist.

Wenn die Ausgabe des Linux-Befehls Tabulatorzeichen enthält, setzt Communications Server für Linux vor dem Senden der Ausgabe an NetView jeden Tabulator in ein Leerzeichen um. Weitere Änderungen an der Ausgabe werden nicht vorgenommen.

Wenn Sie einen Befehl absetzen, während ein anderer Befehl ausgeführt wird (d. h., bevor die Nachricht **UNIX COMMAND COMPLETED** empfangen wird), erscheint die folgende Nachricht:

```
= = = COMMAND QUEUED = = =
```

Der zweite Befehl wird in die Warteschlange gestellt und ausgeführt, sobald der vorherige Befehl abgeschlossen ist.

### Befehl abbrechen

UCF bietet eine Methode zum Abbrechen eines in Ausführung befindlichen Befehls an. Mit dieser Methode kann die Ausführung des aktuellen Befehls gestoppt oder ein interaktiver Befehl wie `vi Dateiname`, der nicht ohne weitere Eingabe abgeschlossen werden kann, abgebrochen werden. Die Methode ist mit der Verwendung einer Unterbrechungssequenz wie **Strg+C** zum Stoppen eines auf einem Terminal laufenden Prozesses oder der Verwendung des Linux-Befehls **kill** zum Stoppen des laufenden Prozesses vergleichbar.

Communications Server für Linux bricht nicht nur den gerade in Ausführung befindlichen Befehl ab, sondern auch alle Befehle, die nach diesem in die Warteschlange gestellt wurden.

Die Befehlssyntax ist mit der für den Linux-Befehl identisch. Anstelle des Befehlstextes wird die Zeichenfolge **ux-cancel** eingegeben. Beispiel:

```
runcmd sp=Mein_Servicepunktname, appl=unix, ux-cancel
```

Für alle noch offenen Befehle (den gerade ausgeführten Befehl und alle in die Warteschlange gestellten Befehle) wird die folgende Nachricht angezeigt:

```
= = = UNIX COMMAND CANCELLED = = =
```

Diese Nachricht gibt an, dass die Linux-Shell, in der der Befehl ausgeführt wurde, gestoppt worden ist. Bei Bedarf können weitere Linux-Befehle abgesetzt werden.

Wenn ein Befehl einen Dämonprozess auf dem Linux-Computer startet, kann dieser Prozess nicht mit **ux-cancel** gestoppt werden. Möglicherweise müssen Sie (auf einem Terminal oder mit UCF) den Linux-Befehl **kill** absetzen, um einen solchen Prozess explizit zu stoppen.

Wird **ux-cancel** verwendet, wenn gerade kein UCF-Befehl ausgeführt wird, zeigt UCF die folgende Nachricht an:

```
NO OUTSTANDING COMMANDS
```

In diesem Fall wird der Befehl **ux-cancel** ignoriert. Es sind keine Aktionen erforderlich. Diese Nachricht kann angezeigt werden, wenn der Befehl **ux-cancel** abgesetzt wird, nachdem der vorherige Befehl beendet ist, aber bevor die Nachricht **UNIX COMMAND COMPLETED** empfangen wurde.

### UCF-Sicherheit

Da UCF einem fernen Bediener ermöglicht, Befehle auf dem Linux-Computer abzusetzen und Ausgaben von diesen Befehlen zu empfangen, sind Sicherheitsaspekte von großer Wichtigkeit. Sie müssen beispielsweise gewährleisten, dass der Bediener nicht auf private Informationen zugreifen oder Linux-Befehle, die die Arbeit anderer Benutzer unterbrechen könnten, absetzen kann.

Die Konfiguration von Communications Server für Linux enthält für den UCF-Benutzer einen speziellen Linux-Systembenutzernamen. Dieser muss eine gültige Anmelde-ID auf dem CS-Linux-Computer sein. Alle UCF-Befehle werden mit der ID dieses Benutzers und so auch mit den Zugriffsberechtigungen dieses Benutzers ausgeführt.

Sie sollten die normalen Sicherheitsfeatures von Linux nutzen, um die Befehle einzuschränken, auf die der UCF-Benutzer zugreifen kann, und nur die Befehle zuzulassen, deren Verwendung über UCF sinnvoll erscheint. Beachten Sie bitte die folgenden Richtlinien:

- Der UCF-Benutzername sollte ein Name sein, der nur für UCF verwendet wird. Verwenden Sie keine vorhandene Anmelde-ID, die auch für andere Zwecke genutzt wird. Dadurch wird es einfacher, die Berechtigungen dieses Benutzers auf die für UCF erwünschten einzugrenzen. Außerdem können Sie dadurch mit UCF gestartete Prozesse identifizieren.
- Sie sollten überlegen, ob es nicht sinnvoll ist, die Benutzer und Gruppen einzuschränken, für die der UCF-Benutzer eine Benutzer-ID oder Gruppen-ID ändern kann. UCF-Benutzern sollten insbesondere die folgenden Möglichkeiten verwehrt bleiben:
  - root oder superuser zu werden.
  - Die Gruppen-ID system zu benutzen, mit der auf das Programm **snaadmin** zugegriffen werden kann. (Auf die Funktionen sollte nicht mit UCF, sondern mit der weiter vorn beschriebenen Funktion SPCF zugegriffen werden.)

**UCF verwenden**

---

## Kapitel 10. Client-Server-Systeme mit Communications Server für Linux verwalten

Communications Server für Linux kann als eigenständiges System verwendet werden (alle SNA-Komponenten und Anwendungen befinden sich auf einem Linux-System) oder als Teil einer Client-Server-Domäne. Eine Client-Server-Domäne umfasst Server (SNA-Knoten) und IBM Remote API Clients (die über einen Server Zugriff auf SNA-Konnektivität haben).

In einer Domäne mit mehreren CS-Linux-Servern befindet sich auf einem Server die Master-Kopie der Konfigurationsdatei für die CS-Linux-Domäne. Dieser Server ist der Hauptserver. Die anderen Server in der Domäne können als Ausweichserver definiert werden. Die Domänenkonfigurationsdatei wird auf die Ausweichserver kopiert, wenn diese gestartet werden oder wenn sich die Master-Kopie ändert, so dass sich auf allen Ausweichservern eine Kopie mit den aktuellen Informationen befindet.

Remote API Clients können Computer mit AIX, Linux, Linux für pSeries, Linux für System z oder Microsoft Windows sein.

Server und Clients kommunizieren mit TCP/IP innerhalb der CS-Linux-Domäne. Es werden IPV4- und IPV6-Adressen unterstützt. Ein Client kann bei Bedarf gleichzeitig auf mehrere Server zugreifen und Anwendungen parallel ausführen. Informationen zu den Voraussetzungen für den Netzbetrieb in einer Client-Server-Konfiguration finden Sie im Abschnitt „Voraussetzung für den IP-Netzbetrieb“ auf Seite 139.

Die TCP/IP-Verbindungen zwischen Clients und Servern können durch physische LANs und WANs führen oder entlang virtueller Pfade zwischen Servern, die in der VM ausgeführt werden, verlaufen. In den Handbüchern zu Communications Server für Linux wird für alle diese Verbindungen der Begriff LAN verwendet.

UNIX

Für Remote API Clients unter AIX oder Linux müssen Sie Angaben zum CS-Linux-Netz und zu den CS-Linux-Servern machen. Hinweise zu dieser Funktion sowie Anweisungen für das Aktivieren und Inaktivieren der Software Communications Server für Linux auf Clients finden Sie im Abschnitt „Remote API Clients unter AIX oder Linux verwalten“ auf Seite 159.

Auf einem Server können alle Verwaltungsbefehle abgesetzt werden. Für das Absetzen von Befehlen auf AIX- und Linux-Clients Linux gelten jedoch Einschränkungen.

- Auf einem AIX- oder Linux-Client können Sie jeden **query**- oder **status**-Befehl absetzen.
- Für einige andere Verwaltungsbefehle, die in der Veröffentlichung "Communications Server for Linux Administration Command Reference" definiert sind, ist explizit angegeben, dass Sie von einem IBM Remote API Client abgesetzt werden können. Andernfalls sind diese Befehle nur von einem Server aus verfügbar.

### WINDOWS

Für Windows-Clients müssen Sie Angaben machen, die Communications Server für Linux zum Aktivieren der Clientsoftware verwenden kann. Falls auf dem Windows-Client aufrufbare Transaktionsprogramme enthalten sind, müssen Sie außerdem Informationen zu den Transaktionsprogrammen bereitstellen. Hinweise zu diesen Funktionen sowie Anweisungen für das Aktivieren und Inaktivieren der Software Communications Server für Linux auf einem Windows-Client finden Sie im Abschnitt „Remote API Clients unter Windows verwalten“ auf Seite 143.

Verwaltungsbefehle, die in der Veröffentlichung "Communications Server for Linux Administration Command Reference" definiert sind, können nicht von einem Windows-Client abgesetzt werden.



---

## Client-Server-Konfiguration ändern

Wenn Sie die Software Communications Server für Linux wie in der Veröffentlichung *Einstieg in Communications Server für Linux* beschrieben installieren, wird die Software zunächst im eigenständigen Modus installiert (alle Komponenten befinden sich auf einem Linux-Computer). Falls Sie Communications Server für Linux als Client-Server-System ausführen möchten, können Sie einen Server als Hauptserver und alle weiteren Server als Ausweichserver konfigurieren. (Sie sollten alle Server mit Ausnahme des Hauptservers als Ausweichserver konfigurieren.)

Communications Server für Linux stellt das Befehlszeilenanwendungsprogramm **snaetutil** bereit, mit dem Server in eine Client-Server-Domäne aufgenommen werden können. Setzen Sie dazu auf jedem Server (beginnend beim Hauptserver) den folgenden Befehl ab:

```
sna stop  
snaetutil Hauptservername [Domänenname]  
sna start
```

Die Parameter des Befehls **snaetutil** werden wie folgt verwendet.

#### *Hauptservername*

Der Name des Hauptservers in der Domäne, zu der er gehört. Wenn Sie den Server in eine vorhandene Domäne verschieben, muss dieser Name mit dem Namen eines Hauptservers in dieser Domäne übereinstimmen.

#### *Domänenname*

Der Name der Domäne, zu der der Server gehört. Dieser Parameter ist optional. Wird er nicht angegeben, verwendet Communications Server für Linux den Standarddomänennamen `ibmcs_domain`.

Setzen Sie den folgenden Befehl ab, um alle übrigen Server als Ausweichserver zu konfigurieren. Sie können diesen Schritt auf dem jeweiligen Ausweichserver oder auf dem Hauptserver ausführen. In beiden Fällen muss jedoch die Software Communications Server für Linux auf dem Hauptserver aktiv sein.

```
snaadmin add_backup, backup_name=Servername
```

*Servername* steht hier für den Namen des Servers, der als Ausweichserver hinzugefügt werden soll.

Mit dem Programm **snanetutil** können Sie einen Server auch aus einer Domäne entfernen, um ihn als eigenständiges System auszuführen.

**Anmerkung:** Diese Option sollten Sie nur verwenden, wenn Communications Server für Linux nicht mehr als Client-Server-System, sondern als eigenständiger Knoten ausgeführt werden soll. Wenn Sie alle Server aus einer vorhandenen Domäne entfernen, können die in der Domäne verbliebenen Clients nicht mehr auf SNA-Ressourcen zugreifen.

Verwenden Sie den folgenden Befehl, um einen Server aus seiner Domäne zu entfernen und ihn als eigenständiges System auszuführen:

```
snanetutil -d
```

### Clients in eine andere Domäne verschieben

Mit dem Programm **snanetutil** können Sie Server in andere Client-Server-Domänen verschieben. Falls Sie Clients in andere Domänen verschieben möchten, müssen Sie dazu die Clientkonfiguration ändern.

Verwenden Sie für jeden zu verschiebenden Remote API Client unter Windows das Clientkonfigurationsdienstprogramm. Geben Sie für den Parameter *Domäne* den neuen Domänennamen an. Weitere Informationen hierzu finden Sie im Abschnitt „Konfiguration für den Remote API Client unter Windows“ auf Seite 145.

Setzen Sie für jeden zu verschiebenden Remote API Client unter AIX oder Linux den Eintrag *Domäne* im Abschnitt Configuration der Clientnetzdatendatei auf den neuen Domänennamen. Weitere Informationen hierzu finden Sie im Abschnitt „Netzdatendatei des Clients (sna\_clnt.net)“ auf Seite 160.

---

## Voraussetzung für den IP-Netzbetrieb

Remote API Clients können über TCP/IP mit CS-Linux-Servern und mit HTTPS über einen WebSphere-Server kommunizieren. Ausführliche Informationen zur Verwendung von HTTPS-Verbindungen finden Sie im Abschnitt „HTTPS-Zugriff für Remote API Clients“ auf Seite 142.

Bevor Sie den Remote API Client ausführen können, müssen Sie auf den Clients und Servern in Ihrem Netz TCP/IP-Port-Adressen konfigurieren. Falls Probleme mit den Standard-Port-Zuordnungen auftreten, müssen Sie bestehende Konflikte lösen. Eine diesbezügliche Beschreibung finden Sie im Abschnitt „IP-Port-Nummern konfigurieren“ auf Seite 140.

Vielleicht möchten Sie außerdem Clients so konfigurieren, dass die TCP/IP-Verbindung automatisch getrennt wird, wenn der Client Communications Server für Linux nicht mehr verwendet. Eine Beschreibung dieses Konfigurationsschrittes finden Sie im Abschnitt „Zeitlimit für LAN-Zugriff“ auf Seite 141.

### IPV4- und IPV6-Adressen

Computer in einer Client-Server-Domäne mit Communications Server für Linux können IPV4- oder IPV6-Adressen verwenden. Alle Server in der Domäne müssen jedoch dasselbe Adressformat verwenden (IPV4 oder IPV6).

- Wenn die Server IPV4 verwenden, müssen Clients auch IPV4 verwenden.

## Voraussetzung für den IP-Netzbetrieb

- Wenn die Server IPV6 verwenden, können Clients IPV6 oder IPV4 verwenden.

Weitere Details zur Konfiguration und Verwendung von IPV4- und IPV6-Adressen enthält die Veröffentlichung *Einstieg in Communications Server für Linux*.

## Hostnamen in der Client-Server-Konfiguration

Communications Server für Linux verwendet für die interne Kommunikation zwischen Servern und Clients vollständig qualifizierte IP-Hostnamen. Das lokale System kann diese Namen normalerweise anhand der Netzkonfiguration (z. B. vom DNS) bestimmen. Falls dies nicht möglich ist, sollten Sie einen vollständig qualifizierten Namen (wie `newbox.this.co.uk`) und keinen Aliasnamen (wie `newbox`) verwenden, wenn in der Konfiguration ein Hostname erforderlich ist.

Der lokale Servername für die einzelnen Computer wird der Datei `/etc/hosts` entnommen. Der Eintrag in dieser Datei muss zuerst die IP-Adresse, dann den vollständig qualifizierten Namen und zum Schluss den Aliasnamen enthalten. Beispiel:

```
9.42.108.28    newbox.this.co.uk    newbox
```

Wenn der Server ein Multi-Home-Server ist (wenn er beispielsweise zwei oder mehr TCP/IP-Netzschnittstellen hat und so mit verschiedenen IP-Adressen erscheint), müssen die Einträge in der Datei `/etc/hosts` für alle Adressen denselben IP-Namen angeben, damit der Name für alle Netzschnittstellen ordnungsgemäß aufgelöst werden kann. Beispiel:

```
9.42.108.28    newbox.this.co.uk    newbox
9.42.80.127    newbox.this.co.uk    newbox
```

Außerdem müssen Sie die Ziele `multi on` in die Datei `/etc/host.conf` aufnehmen, um anzugeben, dass es sich um einen Multi-Home-Server handelt.

## IP-Port-Nummern konfigurieren

Communications Server für Linux verwendet zum Senden von Client-Server-Daten im LAN die TCP/IP- und die UDP/IP-Übertragung. Standardmäßig verwendet die Software für beide Arten der Übertragung die Port-Nummer 1553. Diese Port-Nummer ist für die meisten Installationen geeignet und muss nicht geändert werden.

Sollten beim Aktivieren der Software Communications Server für Linux Probleme auftreten, überprüfen Sie, ob die Fehlerprotokolldatei Nachrichten darüber enthält, dass die von Communications Server für Linux verwendete Port-Nummer mit der von einem anderen Programm genutzten Port-Nummer in Konflikt geraten ist. Liegt eine solche Nachricht vor, führen Sie die folgenden Schritte aus:

1. Überprüfen Sie die Datei `/etc/services` auf dem Computer, auf dem der Fehler aufgetreten ist, um festzustellen, ob ein anderes aufgelistetes Programm die Port-Nummer 1553 für die TCP/IP- oder die UDP/IP-Übertragung verwendet. Ist dies der Fall, versuchen Sie zunächst, dem anderen Programm einen neuen Port zuzuordnen.
2. Wenn das nicht möglich ist oder kein aufgelistetes Programm den Port 1553 verwendet, suchen Sie eine Port-Nummer, die von keinem der Programme, die in der Datei aufgeführt sind, verwendet wird. Überprüfen Sie die Datei `/etc/services` auf allen CS-Linux-Computern in dieser Domäne, um sicherzustellen, dass die Nummer auf keinem der anderen Computer verwendet wird.
3. Fügen auf jedem Computer in der Domäne zwei Zeilen im folgenden Format zur Datei `/etc/services` hinzu:

```
sna-cs      nnnn/tcp
sna-cs      nnnn/udp
```

Der Eintrag *nnnn* steht hier für die neue Port-Nummer. Diese muss auf allen Computern in der CS-Linux-Domäne auf denselben Wert gesetzt werden.

4.

WINDOWS

Falls Ihre CS-Linux-Domäne Windows-Clients enthält, fügen Sie die beiden gleichen Zeilen zur Datei **services** auf jedem der Windows-Computer hinzu. Die Datei **services** hat dasselbe Format wie die Linux-Datei und befindet sich normalerweise im Ausgangsverzeichnis der Windows-TCP/IP-Software. Bei Bedarf finden Sie in der Windows-TCP/IP-Dokumentation weitere Informationen hierzu.

5. Reaktivieren Sie den CS-Linux-Server und die Software Remote API Client.

**Anmerkung:** Um den Port 1553 des Servers (oder die neue Port-Nummer, die Sie für die Client-Server-Kommunikation angegeben haben) vor unbefugtem Zugriff zu schützen, sollten Sie eine Firewall verwenden. Zu und von anderen CS-Linux-Servern und Remote API Clients sollte TCP- und UDP-Datenverkehr zugelassen werden. Andere Computer sollten jedoch nicht auf den Port zugreifen dürfen.

## Zeitlimit für LAN-Zugriff

Wenn der Client mit CS-Linux-Servern über ein Netz kommuniziert, in dem die Verbindungen gebührenpflichtig sind, sollten Sie sicherstellen, dass die TCP/IP-Verbindung vom Client automatisch getrennt wird, sobald die Anwendungen auf dem Client keine CS-Linux-Ressourcen mehr nutzen. Damit wird nicht automatisch die SNA-Software auf dem Client inaktiviert. Sie bleibt aktiv und versucht, die Verbindung zu einem Server wiederherzustellen, falls eine Anwendung diese später benötigt.

Mit dem Parameter *lan\_access\_timeout* (für einen Remote API Client unter AIX oder Linux in der Datei **sna\_clnt.net** und für einen Remote API Client unter Windows in der Registrierungsdatenbank) können Sie die SNA-Software auf dem Client inaktivieren. Die TCP/IP-Verbindung wird getrennt, wenn in der angegebenen Zeit auf dem Client keines der folgenden Ereignisse eingetreten ist:

- Aktive APPC- oder CPI-C-Dialoge (oder Versuche, einen Dialog zu starten)
- Aktive LUA-Sitzungen
- CSV-Verben TRANSFER\_MS\_DATA von einem Windows-Client
- MS-Verben (nur Linux-Clients)
- NOF-Verben (mit Ausnahme der Verben **query\_central\_logger** und **query\_node\_all**)
- Verwaltungsbefehle (mit Ausnahme der folgenden Ereignisse, bei denen der Client die Verbindung **nicht** erneut startet):
  - vom Client protokollierte Fehler- oder Prüfnachrichten (die auch bei Verwendung der zentralen Protokollierung lokal auf dem Client erfasst werden)
  - die Verwaltungsbefehle **query\_central\_logger** und **query\_node\_all** (die Informationen zurückgeben, die vor dem Trennen der TCP/IP-Verbindung verfügbar waren und ggf. nicht mehr mit dem aktuellen Status des LAN übereinstimmen)

## Voraussetzung für den IP-Netzbetrieb

- die NOF-Verben **query\_central\_logger** und **query\_node\_all** (wie bei den entsprechenden Verwaltungsbefehlen)

Die TCP/IP-Verbindung wird insbesondere dann getrennt, wenn Sie die SNA-Software aktivieren, ohne jedoch innerhalb des angegebenen Zeitlimits eine CS-Linux-Anwendung zu starten.

Tritt eines dieser Ereignisse ein, während die TCP/IP-Verbindung inaktiv ist, versucht der Client, die Verbindung zum Server wiederherzustellen, wie es im Abschnitt „Netzdatendatei des Clients (sna\_clnt.net)“ auf Seite 160 bzw. im Abschnitt „Servers“ auf Seite 149 für die Parameter \* und *Servername* beschrieben ist.

Ankommende Attach-Anforderungen für aufgerufene TPs auf diesem Client können nicht akzeptiert werden, solange die TCP/IP-Verbindung inaktiv ist. Die Attach-Anforderung wird zurückgewiesen, als wäre das Zielsystem inaktiv. Dies bedeutet, dass automatisch gestartete TPs auf dem Client nicht verfügbar sind, wenn keine anderen Anwendungen auf dem Client aktiv sind und die TCP/IP-Verbindung das Zeitlimit überschritten hat. Vom Bediener gestartete TPs auf dem Client können jedoch immer verwendet werden, weil ein vom TP abgesetztes Verb *Receive\_Allocate* die TCP/IP-Verbindung wiederherstellt.

---

## HTTPS-Zugriff für Remote API Clients

Wenn Sie mit einem Client-Server-System arbeiten, in dem Remote API Clients über HTTPS eine Verbindung zu CS-Linux-Servern herstellen, benötigen Sie einen Computer, auf dem WebSphere Application Server ausgeführt wird, um diesen Clients den HTTPS-Zugriff auf die Server zu ermöglichen. Anweisungen für das Installieren und Konfigurieren dieses Servers finden Sie in der Veröffentlichung *Einstieg in Communications Server für Linux*.

Wenn Sie neue Server zur CS-Linux-Domäne hinzufügen und Remote API Clients in der Lage sein sollen, per HTTPS auf diese Server zuzugreifen, müssen Sie die WebSphere-Serverkonfigurationsdatei aktualisieren. Nehmen Sie diese Server in die Konfigurationsdatei auf. Die Datei hat den Namen **snahttpsrv.cfg** und befindet sich auf dem WebSphere-Server in dem von der Umgebungsvariablen **USER\_INSTALL\_ROOT** angegebenen Verzeichnis. Falls Sie nicht sicher sind, wo sich dieses Verzeichnis befindet, führen Sie die folgenden Schritte aus.

1. Starten Sie die WebSphere-Administrationskonsole.
2. Wählen Sie in der Menüleiste der Administrationskonsole 'Umgebung' -> 'WebSphere-Variablen verwalten' aus.
3. Suchen Sie in der Liste nach der Variablen **USER\_INSTALL\_ROOT** und notieren Sie den Wert der Variablen. (Dies ist der Pfad eines Verzeichnisses auf dem WebSphere-Server.) Die Liste der Umgebungsvariablen kann zwei oder mehr Seiten umfassen, so dass Sie unter Umständen die Taste zum Vorblättern verwenden müssen, um die gesamte Liste durchzublättern.

Bearbeiten Sie die Konfigurationsdatei in einem Texteditor. Nehmen Sie alle CS-Linux-Server auf, auf die Remote API Clients mit HTTPS zugreifen können. Jeder Server muss in einer gesonderten Zeile der Datei und im folgenden Format angegeben werden:

```
server=servername.domänenname.com
```

Auf jedem Remote API Client, der Zugriff auf den neuen Server haben soll, müssen Sie den Servernamen zur Liste der Server in der Clientnetzdatendatei (bzw. in der Windows-Registrierungsdatenbank für einen Windows-Client) hinzufügen. Lesen Sie hierzu den Abschnitt zum jeweiligen Clienttyp weiter hinten in diesem Kapitel.

---

### Remote API Clients unter Windows verwalten

#### WINDOWS

Mit Communications Server für Linux können Maschinen, auf denen Microsoft Windows ausgeführt wird, in der CS-Linux-Domäne als Clients verwendet werden. Die Software Communications Server für Linux Client enthält API-Bibliotheken, die mit Microsoft Host Integration Server, der Windows Open Systems Architecture (WOSA) und den Schnittstellen von IBM Personal Communications sowie von Communications Server für Windows kompatibel sind. Damit können Anwendungen, die für diese Implementierungen geschrieben wurden, unverändert auf dem Remote API Client unter Windows ausgeführt werden.

Der Remote API Client unter Windows unterstützt die folgenden WOSA-APIs:

- Windows APPC
- Windows CPI-C
- Windows LUA
- Windows CSV

Weitere Informationen zu Windows-SNA-APIs finden Sie in der zu Microsoft Host Integration Server gelieferten Dokumentation.

SNA-Netzdaten und andere für den Remote API Client unter Windows erforderliche Informationen werden in der Windows-Registrierungsdatenbank gespeichert.

Der Client muss aktiviert werden, bevor CS-Linux-Anwendungen oder Emulationsprogramme auf dem Client ausgeführt werden können. Weitere Informationen hierzu finden Sie im Abschnitt „Remote API Client unter Windows aktivieren“ auf Seite 144. Nachdem der Client aktiviert wurde, stellt er über das TCP/IP-Netz eine Verbindung zu einem Server mit Communications Server für Linux her, um auf CS-Linux-Features zugreifen zu können.

Der Betrieb des Clients wird außerdem von den Informationen in der Windows-Registrierungsdatenbank gesteuert. Die Registrierungsdatenbank von Windows enthält Informationen zu Folgendem:

- spezifische Konfigurationsdaten für Remote API Clients unter Windows
- Server, auf die der Client zugreifen kann
- Protokollierungs- und Trace-Optionen für die auf dem Client ausgeführten Anwendungen
- zusätzliche Optionen für CPI-C- und CSV-Anwendungen, die auf dem Client ausgeführt werden
- aufrufbare TPs (APPC oder CPI-C), die auf dem Client ausgeführt werden

Die am häufigsten verwendeten Parameter können Sie auch mit dem Clientkonfigurationsdienstprogramm modifizieren. Dieses Programm sollte bevorzugt für das

## Remote API Clients unter Windows verwalten

Ändern der Parameter verwendet werden. Weitere Informationen hierzu finden Sie im Abschnitt „Konfiguration für den Remote API Client unter Windows“ auf Seite 145.

**Anmerkung:** Falls der Client für den Serverzugriff HTTPS verwendet, müssen Sie die Clientkonfiguration ändern. Geben Sie die Namen der Server an und den WebSphere-Server, der den HTTPS-Zugriff ermöglicht, bevor Sie den Client verwenden. Weitere Informationen hierzu finden Sie im Abschnitt „Konfiguration für den Remote API Client unter Windows“ auf Seite 145.

## Remote API Client unter Windows aktivieren

Der Remote API Client unter Windows kann als ein Windows-Dienst ausgeführt werden. Das Installationsprogramm konfiguriert den Client so, dass er automatisch beim Start des Computers gestartet wird. Sie können den Dienst bei Bedarf auch mit einer der folgenden Methoden manuell starten.

- Starten Sie den Clientdienst in der Systemsteuerung unter "Verwaltung" über das Applet "Dienste".
- Geben Sie in dem mit "Start" > "Ausführen" aufgerufenen Befehlsfenster `net start sxclient` ein.

Der Client sucht dann anhand der Informationen in der Windows-Registrierungsdatenbank, die mit dem Clientkonfigurationsdienstprogramm definiert wurden und im Abschnitt „Konfiguration für den Remote API Client unter Windows“ auf Seite 145 beschrieben sind, nach einem Server, auf dem Communications Server für Linux ausgeführt wird.

**Anmerkung:** Wenn Sie mit Microsoft Windows Vista arbeiten, müssen Sie an einer Eingabeaufforderung mit Administratorberechtigung `net start sxclient` ausführen. Rufen Sie diese Eingabeaufforderung auf, indem Sie mit der rechten Maustaste auf das Symbol 'Eingabeaufforderung' klicken und an der Eingabeaufforderung das Administrator Kennwort eingeben.

## Status eines Remote API Client unter Windows anzeigen

Der Clientmonitor stellt ein Symbol in die Task-Leiste, das den Clientstatus anzeigt, wenn Sie den Mauszeiger auf das Symbol bewegen. Der Monitor ist so konfiguriert, dass er nach dem Start des Computers automatisch ausgeführt wird. Sie können ihn aber auch manuell über das Menü "Start" ausführen. Der Client kann einen der folgenden Status haben:

### **Nicht aktiv**

Der Client wurde nicht gestartet.

### **Keine Verbindung**

Der Client wurde gestartet, hat jedoch noch keine Verbindung zu einem Server hergestellt (oder die Verbindung zum Server wurde unterbrochen).

### *Servername*

Der Client ist mit dem benannten Server verbunden.

## Remote API Client unter Windows inaktivieren

Stellen Sie vor dem Inaktivieren des Clients sicher, dass auf dem Client alle CS-Linux-Anwendungen (3270- und 5250-Emulationsprogramme oder Anwendungen, die CS-Linux-APIs verwenden) gestoppt wurden.

Inaktivieren Sie den Client, indem Sie den Clientdienst auf eine der folgenden Arten stoppen.

- Stoppen Sie den Clientdienst in der Systemsteuerung unter "Verwaltung" über das Applet "Dienste".
- Geben Sie in dem mit "Start" > "Ausführen" aufgerufenen Befehlsfenster `net stop sxclient` ein.

Auf einem Computer mit Windows-Terminaldiensten bedeutet dies, dass Benutzer den Client nicht mehr verwenden können.

**Anmerkung:** Wenn Sie mit Microsoft Windows Vista arbeiten, müssen Sie an einer Eingabeaufforderung mit Administratorberechtigung `net stop sxclient` ausführen. Rufen Sie diese Eingabeaufforderung auf, indem Sie mit der rechten Maustaste auf das Symbol 'Eingabeaufforderung' klicken und an der Eingabeaufforderung das Administrator Kennwort eingeben.

### Konfiguration für den Remote API Client unter Windows

Die Konfigurationsdaten von Remote API Clients unter Windows werden in der Windows-Registrierungsdatenbank gespeichert. Die Registrierungsdatenbank von Windows enthält SNA-Netzdaten (ähnlich den Daten in der Clientnetzdatendatei auf Remote API Clients unter AIX oder Linux). Außerdem enthält die Datenbank zusätzliche Konfigurationsdaten speziell für Remote API Clients unter Windows.

**Anmerkung:** Die Konfigurationsdaten für eine CPI-C-Anwendung (Name des lokalen TP und Aliasname der lokalen LU) können in Umgebungsvariablen oder in der Registrierungsdatenbank angegeben werden. Die Verwendung von Umgebungsvariablen kann angezeigt sein, wenn Sie Windows Terminal Server verwenden und mehrere Kopien einer Anwendung mit verschiedenen lokalen LUs ausführen müssen. Ausführliche Informationen hierzu finden Sie im Abschnitt „Appl\_Name“ auf Seite 158.

Mit dem Clientkonfigurationsdienstprogramm können die am häufigsten verwendeten Clientkonfigurationsparameter ganz einfach modifiziert werden. Dieses Programm sollte bevorzugt für das Ändern der Parameter genutzt werden. Wenn Sie dieses Programm verwenden möchten, müssen Sie in der Programmgruppe 'IBM Remote API' das **Konfigurationsdienstprogramm** ausführen. Das Programm zeigt das gleiche Konfigurationsfenster wie beim Installationsprozess an. Weitere Informationen zum Modifizieren der Parameter finden Sie im Kapitel 'Remote API Client unter Windows installieren' der Veröffentlichung "Einstieg in Communications Server für Linux".

**Anmerkung:** Wenn Sie die Clientkonfigurationsparameter geändert haben, müssen Sie den Client stoppen und neu starten, um die Änderungen in Kraft zu setzen. Ausführliche Informationen hierzu finden Sie in den Abschnitten „Remote API Client unter Windows inaktivieren“ auf Seite 144 und „Remote API Client unter Windows aktivieren“ auf Seite 144.

Die Daten werden in der Registrierungsdatenbank in Form von Werten gespeichert, die unter Teilschlüsseln des folgenden Schlüssels konfiguriert werden:

```
\\HKEY_LOCAL_MACHINE\SOFTWARE\SNA Client\SxClient\Parameters
```

## Remote API Clients unter Windows verwalten

Für die einzelnen Teilschlüssel der Registrierungsdatenbank sind die folgenden Werte gültig:

### Configuration

```
domain = Domänenname
  maximum_process_count = nn
  maximum_header_count = nn
  maximum_element_count = nn
invoked_tps = YES | NO
lan_access_timeout = nn
broadcast_attempt_count = nn
server_lost_timeout = nn
  client_start_timeout = nn
```

### Servers

```
Server1 = * | [ Webservername : [ Port-Nummer : ] ] Servername1
Server2 = [ Webservername : [ Port-Nummer : ] ] Servername2
.
.
.
Server9 = [ Webservername : [ Port-Nummer : ] ] Servername9
```

### Logging

```
exception_logging_enabled = YES | NO
audit_logging_enabled = YES | NO
log_directory = Verzeichnis
error_file = Name_der_Fehlerdatei
backup_error_file = Name_der_Sicherungsfehlerdatei
error_file_wrap_size = Größe_der_Fehlerdatei
audit_file = Name_der_Prüfdatei
backup_audit_file = Name_der_Sicherungsprüfdatei
audit_file_wrap_size = Größe_der_Prüfdatei
succinct_errors = YES | NO
succinct_audits = YES | NO
```

### API\_tracing

```
file1 = Name_der_Trace-Datei_1
file2 = Name_der_Trace-Datei_2
flip_size = Dateigröße
truncation_length = Länge
all_api = YES | NO
appc = YES | NO
cpic = YES | NO
csv = YES | NO
rui = YES | NO
nof = YES | NO
```

### CS\_tracing

```
file1 = Name_der_CS-Trace-Datei_1
file2 = Name_der_CS-Trace-Datei_2
flip_size = Dateigröße
admin_msg = YES | NO
datagram = YES | NO
data = YES | NO
send = YES | NO
receive = YES | NO
```

### Internal\_tracing

```
file1 = Name_der_internen-Trace-Datei_1
file2 = Name_der_internen-Trace-Datei_2
flip_size = Dateigröße
trace_level = nn
trace_flushing = YES | NO
```

### AppL\_Name

```
APPCTPN = TP-Name
```

APPCLLU = *LU-Name*

CSV\_data  
CSVTLG = *Dateiname\_der\_Tabelle\_G*

**Anmerkung:** Der Wert *domain = Domänenname* ist der einzige erforderliche Wert in der Registrierungsdatenbank.

Der folgende Abschnitt erläutert die einzelnen Konfigurationsparameter. Wenn ein Parameter den Wert YES oder NO annehmen kann, wird jede mit Y oder y beginnende Zeichenfolge als YES und jede mit N oder n beginnende Zeichenfolge als NO interpretiert.

### Configuration

Der Teilschlüssel Configuration enthält die folgenden Konfigurationsdaten für den Client:

*domain* Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Der Wert *Domänenname* gibt den Domännennamen für das CS-Linux-LAN an, wie er bei der Clientinstallation festgelegt wurde. Diese Zeile ist erforderlich.

*maximum\_process\_count*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Geben Sie die maximale Gesamtanzahl APPC-, CPI-C-, LUA- und NOF-Anwendungen an, die gleichzeitig auf diesem Client ausgeführt werden können.

Dieser Parameter ist optional. Der Standardwert liegt bei 240 und ist normalerweise ausreichend. Falls Sie Fehlermeldungen sehen, die darauf hinweisen, dass ein IPC-Steuerblock nicht zugeordnet werden konnte, müssen Sie möglicherweise die maximale Prozessanzahl erhöhen, indem Sie diesen Parameter angeben. Der größtmögliche Wert ist 4096.

*maximum\_header\_count, maximum\_element\_count*

Diese Werte haben in der Registrierungsdatenbank den Datentyp REG\_SZ.

Diese beiden Parameter sind optional. Die Standardwerte liegen bei 1250 und 1800, was normalerweise ausreichend ist. Sofern Sie nicht von der Benutzerunterstützung dazu aufgefordert werden, müssen Sie in der Regel keine Werte für diese Parameter angeben.

*invoked\_tps*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Geben Sie einen der folgenden Werte an:

**YES** Auf diesem Client werden aufgerufene Transaktionsprogramme ausgeführt (APPC-TPs, die RECEIVE\_ALLOCATE absetzen, oder CPI-C-Anwendungen, die Accept\_Conversation oder Accept\_Incoming absetzen). In diesem Fall müssen Sie unter Umständen auch das Transaktionsprogramm auf diesem Client definieren. Weitere Informationen hierzu finden Sie im Abschnitt „TPs definieren“ auf Seite 93 oder in Anhang B, „Aufrufbares TP mit Befehlszeilendienstprogramm konfigurieren“, auf Seite 189.

**NO** Der Client wird nicht für die Ausführung aufgerufener Transaktionsprogramme verwendet.

Diese Zeile ist optional. Ist sie nicht angegeben, wird die Standardeinstellung NO verwendet.

## Remote API Clients unter Windows verwalten

### *lan\_access\_timeout*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Geben Sie die Zeit in Sekunden an, die die IP- oder HTTPS-Verbindung vom Client zu einem Server aktiv bleiben soll, wenn keine Anwendungen auf dem Client CS-Linux-Ressourcen verwenden. Weitere Informationen hierzu finden Sie im Abschnitt „Zeitlimit für LAN-Zugriff“ auf Seite 141.

Der gültige Bereich ist 0–65535. Das Mindestzeitlimit liegt bei 60 Sekunden (niedrigere Werte werden auf 60 Sekunden hochgesetzt). Wenn Sie die Verbindung schneller inaktivieren möchten, inaktivieren Sie den Client.

Dieser Parameter ist optional. Ist er nicht angegeben, wird standardmäßig kein Zeitlimit verwendet, so dass die Verbindung aktiv bleibt, solange der Client ausgeführt wird.

### *broadcast\_attempt\_count*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Wenn der Client mit der Broadcast-Methode Kontakt zu einem Server aufnimmt (wie von dem im Abschnitt „Servers“ auf Seite 149 beschriebenen Parameter \* angegeben), gibt dieser Parameter die maximale Anzahl von Broadcasts für einen Verbindungsversuch zu einem Server an.

Der gültige Bereich ist 1–65535. Der Mindestwert ist 1. Wenn Sie einen höheren Wert angeben, wiederholt der Client seinen Versuch alle 10 Sekunden, bis der Kontakt zu einem Server hergestellt oder dieser Maximalwert erreicht ist. Wird der Maximalwert erreicht, ohne dass eine Verbindung zu einem Server hergestellt wurde, versucht der Client, Kontakt zu einem benannten Server aufzunehmen (wie im Abschnitt „Servers“ auf Seite 149 beschrieben).

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung 5 verwendet.

### *server\_lost\_timeout*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Wenn der Client den Kontakt zu einem Server verliert und die Verbindung erneut herstellen muss oder wenn er weder im Broadcast-Betrieb noch unter Verwendung eines benannten Servers Kontakt zu einem Server aufnehmen kann (wie im Abschnitt „Servers“ auf Seite 149 beschrieben), gibt dieser Parameter die Zeit in Sekunden an, die der Client abwartet, bevor er versucht, eine Verbindung zu einem Server herzustellen. Für den Fall, dass der Client den Kontakt zu einem Server verloren hat, wartet Communications Server für Linux nicht diese ganze Zeitspanne, sondern unternimmt nach einer wahlfreien Zeit zwischen fünf Sekunden und dem angegebenen Zeitlimit einen erneuten Versuch. Auf diese Weise sollen Überlastungen beim Datenaustausch im Netz vermieden werden, die durch eine große Anzahl von Clients ausgelöst werden könnten, wenn diese zur selben Zeit versuchen, Kontakt zu einem Server aufzunehmen.

Dieser Parameter ist optional. Der gültige Bereich ist 5–65535. Ist er nicht angegeben, wird die Standardeinstellung 200 (Sekunden) verwendet.

### *client\_start\_timeout*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_DWORD.

Gibt die Zeit in Sekunden an, die eine Anwendung darauf wartet, dass der Client gestartet wird und versucht, Kontakt zu einem Server aufzunehmen. Gültig sind Werte zwischen 0 und 300. Alle Werte außerhalb dieses

Bereichs werden zwingend auf einen Wert innerhalb des Bereichs gesetzt. Der Standardwert liegt bei 10 Sekunden.

Mit diesem Parameter können Ereignisse gesteuert werden, wenn die Anwendung und der Client so konfiguriert sind, dass sie beim Systemstart gestartet werden (weil sie im Startordner enthalten oder ein automatisch gestarteter Dienst sind). Die Anwendung wartet die in diesem Feld angegebene Zeit in Sekunden ab, damit der Client zuerst gestartet werden kann. Dies ermöglicht dem Client, eine Verbindung zu einem Server herzustellen, um die für die Anwendung erforderlichen Ressourcen bereitstellen zu können, bevor die Ausführung der Anwendung wegen mangelnder Ressourcen scheitert.

### Servers

Der Teilschlüssel Servers enthält Informationen zu CS-Linux-Konfigurationsservern, von denen der Client Konfigurationsdaten anfordern soll. Diese Liste sollte die Namen des Hauptkonfigurationsservers und aller Ausweichserver enthalten, die sich in derselben Domäne wie der Client befinden. Informationen zum Konfigurieren von Haupt- und Ausweichservern finden Sie im Abschnitt „Client-Server-Funktionen konfigurieren“ auf Seite 57.

**Anmerkung:** Das Format und die Bedeutung dieses Unterschlüssels hängen davon ab, ob sich der Client in demselben privaten Netz wie die Server befindet oder ob die Verbindung mit HTTPS über ein öffentliches Netz hergestellt werden muss.

#### *Server1*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Geben Sie einen Stern (\*) oder einen Servernamen ein.

- Wenn der Client sich in demselben privaten Netz wie seine Server befindet und mit einer UDP-Broadcast-Nachricht an alle Computer in seinem TCP/IP-Teilnetz (oder an alle erreichbaren Teilnetze, wenn der Client-computer mehr als eine LAN-Adapterkarte enthält) versuchen soll, einen Server mit CS Linux zu finden, geben Sie \* an.

Diese Option ist nur verfügbar, wenn der Client IPV4-Adressen verwendet. UDP-Broadcasts werden für IPV6 nicht unterstützt.

Der Client wiederholt den Broadcast-Versuch alle 10 Sekunden, bis die vom Parameter *broadcast\_attempt\_count* angegebene Anzahl von Versuchen erreicht ist oder der Kontakt zu einem Server hergestellt wurde. Wird die von *broadcast\_attempt\_count* angegebene Begrenzung erreicht, bevor der Kontakt zu einem Server hergestellt werden kann, versucht der Client, direkte Nachrichten an benannte Server (die von den folgenden Zeilen der Datei angegeben werden) zu senden.

- Geben Sie für Fälle, in denen der Client sich in demselben privaten Netz wie die Server befindet, jedoch keinen Server mit UDP-Broadcasts erreichen kann und direkte Nachrichten senden muss, den Namen des ersten Servers an, mit dem der Client eine Kontaktaufnahme versuchen soll. Die Parameter *Webserververname* und *Port-Nummer* werden nicht verwendet und sollten nicht angegeben werden. Dies gilt in den folgenden Fällen:
  - Wenn der Client IPV6-Adressen verwendet (UDP-Broadcasts werden nicht unterstützt).
  - Das CS-Linux-LAN umfasst mehrere TCP/IP-Teilnetze und nicht alle TCP/IP-Teilnetze, die der Client mit UDP erreichen kann, enthalten CS-Linux-Server.
  - Auf dem Client ist keine UDP-Unterstützung installiert.

## Remote API Clients unter Windows verwalten

In anderen Fällen ist die Verwendung von UDP-Broadcasts optional. Wenn keine Broadcast-Versuche unternommen werden sollen, geben Sie an Stelle von \* den Namen des ersten Servers an.

- Wenn der Client mit HTTPS auf die Server zugreift, werden UDP-Broadcasts nicht unterstützt. Geben Sie in diesem Fall den Namen des WebSphere-Servers, der die HTTPS-Unterstützung bereitstellt, und den Namen des CS-Linux-Servers im folgenden Format an:

*Webservername : Servername1*

Hierfür wird vorausgesetzt, dass WebSphere für die Verwendung von HTTPS-Verbindungen am Standard-Port 443 konfiguriert ist. Falls Ihr Netzadministrator WebSphere für die Verwendung eines anderen Ports konfiguriert hat, nehmen Sie die Port-Nummer im folgenden Format auf:

*Webservername : Port-Nummer : Servername1*

Ausführliche Informationen zum Konfigurieren von WebSphere für die Unterstützung von HTTPS-Verbindungen finden Sie in der Veröffentlichung *Einstieg in Communications Server für Linux*.

*Server2 – Server9*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Geben Sie die Namen zusätzlicher CS-Linux-Server in der Reihenfolge an, in der der Client versuchen soll, eine Verbindung herzustellen. Verwenden Sie dasselbe Format wie für *Server1*.

Wenn der Client versucht hat, mit UDP-Broadcasts einen Server zu erreichen (oder versucht hat, Kontakt zu dem unter *Server1* angegebenen Server aufzunehmen), ohne eine Antwort zu empfangen, versucht der Client, den als *Server2* angegebenen Server mit einer direkten Nachricht zu erreichen. Schlägt dieser Versuch fehl, versucht der Client es mit dem als *Server3* angegebenen Server usw. Diese Servernamen sind optional, stellen aber einen Ausweichmechanismus dar, wenn die Broadcast-Methode für die Kontaktaufnahme mit einem Server scheitert oder der als *Server1* angegebene Server nicht verfügbar ist.

Ist der Versuch des Clients, zu jedem der aufgelisteten Server Kontakt aufzunehmen, erfolglos, wartet der Client die vom Parameter *server\_lost\_timeout* angegebene Zeit und versucht dann erneut, eine Verbindung zu einem Server herzustellen (mit UDP-Broadcasts oder dem ersten Server in der Liste).

Die Parameter *Server2 – Server9* können nicht auf \* gesetzt werden, um die Verwendung von UDP-Broadcasts anzugeben. Hierfür kann nur der Parameter *Server1* genutzt werden, weil der Wert \* vor allen Servernamen in der Datei stehen muss.

### Logging

Der Teilschlüssel Logging gibt die Protokollierungsoptionen für den Client an. Mit diesen Optionen können die Protokollierungseinstellungen für den Client angegeben werden, die die für die gesamte Domäne festgelegten Protokollierungsoptionen außer Kraft setzen. Weitere Informationen zum Angeben der Protokollierungsoptionen für eine Domäne finden Sie im Abschnitt „Protokollierung konfigurieren“ auf Seite 60.

Wenn die zentrale Protokollierung aktiviert ist, werden alle Protokollnachrichten in eine zentrale Datei auf einem Server geschrieben. In diesem Fall werden nur die hier angegebenen Parameter *exception\_logging\_enabled* und *audit\_logging\_enabled* verwendet. Alle anderen Parameter werden ignoriert.

Die Protokollierungsoptionen werden wie folgt angegeben:

### *exception\_logging\_enabled*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Setzen Sie diesen Parameter auf einen der folgenden Werte:

**YES** Ausnahmenachrichten werden erfasst.

**NO** Es werden keine Ausnahmenachrichten erfasst.

Dieser Parameter ist optional. Wenn er nicht angegeben ist, bestimmt der Client anhand der globalen Domäneneinstellungen, ob Ausnahmenachrichten erfasst werden sollen. (Die Ausgangseinstellung ist standardmäßig YES, so dass Ausnahmenachrichten erfasst werden.)

### *audit\_logging\_enabled*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Setzen Sie diesen Parameter auf einen der folgenden Werte:

**YES** Prüfnachrichten werden erfasst.

**NO** Es werden keine Prüfnachrichten erfasst.

Dieser Parameter ist optional. Wenn er nicht angegeben ist, bestimmt der Client anhand der globalen Domäneneinstellungen, ob Prüfnachrichten erfasst werden sollen. (Die Ausgangseinstellung ist standardmäßig YES, so dass Prüfnachrichten erfasst werden.)

### *log\_directory*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Der vollständige Pfad zu dem Verzeichnis, in dem Protokolldateien auf diesem Client gespeichert werden. Alle (mit den folgenden Parametern angegebenen) Protokolldateien und Sicherungsprotokolldateien werden in diesem Verzeichnis gespeichert. Wenn Sie die im *Communications Server for Linux Diagnostics Guide* beschriebene Protokollfilterfunktion verwenden, wird die Datei **logfilter.txt** (die diese Funktion steuert) ebenfalls in diesem Verzeichnis gespeichert.

Dieser Parameter ist optional. Ist er nicht angegeben, werden die Dateien im Installationsverzeichnis von Windows gespeichert.

### *error\_file*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Name der Datei, in die Fehlernachrichten geschrieben werden. Dieser Parameter ist optional. Ist er nicht angegeben, wird der Standardwert **sna.err** verwendet.

Falls Sie Fehler- und Prüfnachrichten in einer Datei erfassen möchten, geben Sie für diesen Parameter und den Parameter *audit\_file* denselben Dateinamen an.

### *backup\_error\_file*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Name der Sicherungsfehlerprotokolldatei. Erreicht die Fehlerprotokolldatei die vom Parameter *error\_file\_wrap\_size* angegebene Größe, kopiert Communications Server für Linux den Inhalt in die Sicherungsdatei (die eine ggf. vorhandene Datei überschreibt) und löscht anschließend den Inhalt der Protokolldatei.

## Remote API Clients unter Windows verwalten

Dieser Parameter ist optional. Ist er nicht angegeben, wird der Standardwert **bak.err** verwendet.

Falls Sie Fehler- und Prüfnachrichten in einer Datei erfassen möchten, geben Sie für diesen Parameter und den Parameter *backup\_audit\_file* denselben Dateinamen an.

### *error\_file\_wrap\_size*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_DWORD.

Die maximale Größe der von *error\_file* angegebenen Protokolldatei. Führt das Schreiben einer Nachricht in die Datei dazu, dass diese Begrenzung überschritten wird, kopiert Communications Server für Linux den aktuellen Inhalt der Protokolldatei in die Sicherungsprotokolldatei und löscht ihn anschließend aus der Protokolldatei. Der maximale Plattenspeicherplatz, der von Fehlerprotokolldateien belegt wird, liegt demzufolge etwa bei dem Zweifachen des Parameters *error\_file\_wrap\_size*.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung 1000000 (Bytes) verwendet. Wenn Sie Fehler- und Prüfnachrichten in derselben Datei protokollieren, muss dieser Parameter auf denselben Wert wie der Parameter *audit\_file\_wrap\_size* gesetzt werden.

### *audit\_file*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Name der Datei, in die Prüfnachrichten geschrieben werden. Dieser Parameter ist optional. Ist er nicht angegeben, wird der Standardwert **sna.aud** verwendet.

Falls Sie Fehler- und Prüfnachrichten in einer Datei erfassen möchten, geben Sie für diesen Parameter und den Parameter *error\_file* denselben Dateinamen an.

### *backup\_audit\_file*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Name der Sicherungsprüfprotokolldatei. Erreicht die Prüfprotokolldatei die vom Parameter *audit\_file\_wrap\_size* angegebene Größe, kopiert Communications Server für Linux den Inhalt in die Sicherungsdatei (die eine ggf. vorhandene Datei überschreibt) und löscht anschließend den Inhalt der Prüfprotokolldatei.

Dieser Parameter ist optional. Ist er nicht angegeben, wird der Standardwert **bak.aud** verwendet.

Falls Sie Fehler- und Prüfnachrichten in einer Datei erfassen möchten, geben Sie für diesen Parameter und den Parameter *backup\_error\_file* denselben Dateinamen an.

### *audit\_file\_wrap\_size*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_DWORD.

Die maximale Größe der von *audit\_file* angegebenen Protokolldatei. Führt das Schreiben einer Nachricht in die Datei dazu, dass diese Begrenzung überschritten wird, kopiert Communications Server für Linux den aktuellen Inhalt der Protokolldatei in die Sicherungsprotokolldatei und löscht ihn anschließend aus der Protokolldatei. Der maximale Plattenspeicherplatz, der von Prüfprotokolldateien belegt wird, liegt demzufolge etwa bei dem Zweifachen des Parameters *audit\_file\_wrap\_size*.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung 1000000 (Bytes) verwendet. Wenn Sie Fehler- und Prüfnachrichten

in derselben Datei protokollieren, muss dieser Parameter auf denselben Wert wie der Parameter *error\_file\_wrap\_size* gesetzt werden.

### *succinct\_errors*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Gibt an, ob in der Fehlerprotokolldatei die kurze oder ausführliche Protokollierung angewendet werden soll. Diese Einstellung gilt sowohl für Ausnahmeprotokolle als auch für Fehlerprotokolle. Sie können einen der folgenden Werte angeben:

- YES** Es wird die kurze Protokollierung verwendet. Jede Nachricht in der Protokolldatei enthält eine Zusammenfassung des Nachrichtenkopfs (z. B. die Nachrichtennummer und den Protokolltyp) sowie den Nachrichtentext und die Parameter. Weitere Details zur Ursache der Protokollnachricht und ggf. erforderliche Maßnahmen können Sie mit dem Dienstprogramm **snahelp** auf einem Linux-Computer abrufen.
- NO** Es wird die ausführliche Protokollierung verwendet. Jede Nachricht in der Protokolldatei enthält eine vollständige Auflistung der Informationen im Nachrichtenkopf, den Nachrichtentext und die Parameter sowie zusätzliche Angaben zur Ursache der Protokollnachricht und zu erforderlichen Maßnahmen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird der Standardwert von dem Befehl **set\_global\_log\_type** übernommen, der zuletzt an den Hauptserver abgesetzt (oder mit dem Motif-Verwaltungsprogramm festgelegt) wurde. Vor dem Absetzen eines Befehls **set\_global\_log\_type** wird als Ausgangseinstellung standardmäßig die kurze Protokollierung verwendet.

Wenn Sie mit der zentralen Protokollierung arbeiten, richtet sich die Verwendung der kurzen oder ausführlichen Protokollierung für Nachrichten aller Computer nach der Einstellung dieses Parameters auf dem Server, der als zentrale Protokollstelle fungiert. Diese Einstellung kann von dem Befehl **set\_global\_log\_type** oder **set\_log\_type** übernommen worden sein, der an den Server abgesetzt wurde, um die Standardeinstellung außer Kraft zu setzen.

### *succinct\_audits*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Gibt an, ob in der Prüfprotokolldatei die kurze oder ausführliche Protokollierung angewendet werden soll. Die zulässigen Werte und ihre Bedeutung stimmen mit denen für den Parameter *succinct\_errors* überein.

## API\_tracing

Der Teilschlüssel **API\_tracing** gibt die API-Trace-Optionen für Anwendungen an, die auf dem Client ausgeführt werden. Weitere Informationen zum Durchführen von Traces finden Sie im *Communications Server for Linux Diagnostics Guide*. Die Trace-Optionen werden wie folgt angegeben:

*file1* Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Der vollständige Name der Trace-Datei bzw. der ersten Trace-Datei, wenn der Trace in zwei Dateien erfasst wird. (Vergleichen Sie hierzu die Beschreibung des Parameters *file2*.)

Dieser Parameter ist erforderlich, wenn Sie den API-Trace aktivieren möchten.

## Remote API Clients unter Windows verwalten

- file2* Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.  
Der vollständige Pfadname der zweiten Trace-Datei. Dieser Parameter ist optional. Wenn der Trace in einer Datei und nicht in zwei Dateien erfasst werden soll, nehmen Sie diese Zeile nicht auf.  
Sind *file1* und *file2* angegeben, erfolgt die Erfassung des Trace in zwei Dateien. Erreicht die erste Datei die vom Parameter *flip\_size* angegebene Größe, wird der Inhalt der zweiten Datei gelöscht und die Erfassung des Trace in der zweiten Datei fortgesetzt. Wenn diese Datei die von *flip\_size* angegebene Größe erreicht, wird der Inhalt der ersten Datei gelöscht und die Erfassung des Trace in der ersten Datei fortgesetzt. Auf diese Weise wird sichergestellt, dass der Trace lange Zeit fortgesetzt werden kann, ohne übermäßig viel Plattenspeicherplatz zu belegen. Der maximal erforderliche Speicherplatz liegt ungefähr bei dem Zweifachen des Parameters *flip\_size*.
- flip\_size* Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_DWORD.  
Die maximale Größe der Trace-Datei. Wurden zwei Dateinamen angegeben, wechselt der Trace von einer Datei zur nächsten, wenn die aktuelle Datei diese Größe erreicht hat. Wenn nur ein Dateiname angegeben wurde, wird dieser Parameter ignoriert. Die Dateigröße ist nicht begrenzt.  
Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung 1000000 (Bytes) verwendet.
- truncation\_length* Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_DWORD.  
Die maximale Länge der Informationen in Bytes, die für jede Nachricht in die Trace-Datei geschrieben werden. Ist eine Nachricht länger als hier angegeben, schreibt Communications Server für Linux nur den Anfang der Nachricht in die Trace-Datei. Die über *truncation\_length* hinausgehenden Daten werden gelöscht. Auf diese Weise können Sie für jede Nachricht die wichtigsten Informationen erfassen und gleichzeitig vermeiden, dass sich die Datei mit langen Nachrichten füllt.  
Dieser Parameter ist optional. Ist er nicht angegeben, schneidet Communications Server für Linux keine Nachrichten ab. (Für jede Nachricht werden alle Daten in die Datei geschrieben.)
- all\_api* Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.  
Sollen Trace-Nachrichten zu allen APIs erfasst werden, müssen Sie diesen Parameter auf YES setzen. In diesem Fall ignoriert Communications Server für Linux die Parameter *appc* bis *nof*.  
Falls Sie den Trace für alle APIs inaktivieren möchten, setzen Sie *all\_api* und alle Parameter von *appc* bis *nof* auf NO.  
Falls nur Trace-Nachrichten zu bestimmten APIs erfasst werden sollen, setzen Sie *all\_api* auf NO, und geben Sie mit den Parametern *appc* bis *nof* an, für welche APIs ein Trace durchgeführt werden soll.  
Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung NO verwendet.
- appc* Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.  
Wenn Sie Trace-Nachrichten für die API APPC erfassen möchten, setzen Sie diesen Parameter auf YES. Andernfalls müssen Sie ihn auf NO setzen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung `N0` verwendet. Wenn der Parameter `all_api` auf `YES` gesetzt ist, wird dieser Parameter ignoriert und APPC-Trace-Nachrichten werden erfasst.

*cpic* Dieser Wert hat in der Registrierungsdatenbank den Datentyp `REG_SZ`.  
Wenn Sie Trace-Nachrichten für die API CPI-C erfassen möchten, setzen Sie diesen Parameter auf `YES`. Andernfalls müssen Sie ihn auf `N0` setzen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung `N0` verwendet. Wenn der Parameter `all_api` auf `YES` gesetzt ist, wird dieser Parameter ignoriert und CPI-C-Trace-Nachrichten werden erfasst.

*csv* Dieser Wert hat in der Registrierungsdatenbank den Datentyp `REG_SZ`.

Wenn Sie Trace-Nachrichten für die API CSV erfassen möchten, setzen Sie diesen Parameter auf `YES`. Andernfalls müssen Sie ihn auf `N0` setzen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung `N0` verwendet. Wenn der Parameter `all_api` auf `YES` gesetzt ist, wird dieser Parameter ignoriert und CSV-Trace-Nachrichten werden erfasst.

*rui* Dieser Wert hat in der Registrierungsdatenbank den Datentyp `REG_SZ`.

Wenn Sie Trace-Nachrichten für die LUA RUI erfassen möchten, setzen Sie diesen Parameter auf `YES`. Andernfalls müssen Sie ihn auf `N0` setzen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung `N0` verwendet. Wenn der Parameter `all_api` auf `YES` gesetzt ist, wird dieser Parameter ignoriert und LUA-RUI-Trace-Nachrichten werden erfasst.

*nof* Dieser Wert hat in der Registrierungsdatenbank den Datentyp `REG_SZ`.

Wenn Sie Trace-Nachrichten für die API NOF erfassen möchten, setzen Sie diesen Parameter auf `YES`. Andernfalls müssen Sie ihn auf `N0` setzen. NOF-Nachrichten werden nicht direkt von Anwendungen auf Windows-Clients, sondern intern von CS-Linux-Komponenten zum Anfordern von Konfigurationsdaten verwendet.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung `N0` verwendet. Wenn der Parameter `all_api` auf `YES` gesetzt ist, wird dieser Parameter ignoriert und NOF-Trace-Nachrichten werden erfasst.

### CS\_tracing

Der Teilschlüssel `CS_tracing` gibt Optionen für den Client-Server-Trace an (Trace für Nachrichten zwischen dem Client und CS-Linux-Servern). Weitere Informationen zum Durchführen von Traces finden Sie im *Communications Server for Linux Diagnostics Guide*. Die Trace-Optionen werden wie folgt angegeben:

*file1* Dieser Wert hat in der Registrierungsdatenbank den Datentyp `REG_SZ`.

Der vollständige Name der Trace-Datei bzw. der ersten Trace-Datei, wenn der Trace in zwei Dateien erfasst wird. (Vergleichen Sie hierzu die Beschreibung des Parameters *file2*.)

Dieser Parameter ist erforderlich, wenn Sie den Client-Server-Trace aktivieren möchten. Außerdem müssen Sie den Parameter `trace_flags` setzen.

*file2* Dieser Wert hat in der Registrierungsdatenbank den Datentyp `REG_SZ`.

Der vollständige Pfadname der zweiten Trace-Datei. Dieser Parameter ist optional. Wenn der Trace in einer Datei und nicht in zwei Dateien erfasst werden soll, nehmen Sie diese Zeile nicht auf.

## Remote API Clients unter Windows verwalten

Sind *file1* und *file2* angegeben, erfolgt die Erfassung des Trace in zwei Dateien. Erreicht die erste Datei die vom Parameter *flip\_size* angegebene Größe, wird der Inhalt der zweiten Datei gelöscht und die Erfassung des Trace in der zweiten Datei fortgesetzt. Wenn diese Datei die von *flip\_size* angegebene Größe erreicht, wird der Inhalt der ersten Datei gelöscht und die Erfassung des Trace in der ersten Datei fortgesetzt. Auf diese Weise wird sichergestellt, dass der Trace lange Zeit fortgesetzt werden kann, ohne übermäßig viel Plattenspeicherplatz zu belegen. Der maximal erforderliche Speicherplatz liegt ungefähr bei dem Zweifachen des Parameters *flip\_size*.

### *flip\_size*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_DWORD.

Die maximale Größe der Trace-Datei. Wurden zwei Dateinamen angegeben, wechselt der Trace von einer Datei zur nächsten, wenn die aktuelle Datei diese Größe erreicht hat. Wenn nur ein Dateiname angegeben wurde, wird dieser Parameter ignoriert. Die Dateigröße ist nicht begrenzt.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung 1000000 (Bytes) verwendet.

### *admin\_msg*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Wenn Sie einen Trace für interne Nachrichten zur Client-Server-Topologie durchführen möchten, setzen Sie diesen Parameter auf YES. Andernfalls müssen Sie ihn auf NO setzen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung NO verwendet.

### *datagram*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Wenn Sie einen Trace für Datagrammnachrichten durchführen möchten, setzen Sie diesen Parameter auf YES. Andernfalls müssen Sie ihn auf NO setzen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung NO verwendet.

### *data*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Wenn Sie einen Trace für Datennachrichten durchführen möchten, setzen Sie diesen Parameter auf YES. Andernfalls müssen Sie ihn auf NO setzen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung NO verwendet.

### *send*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Wenn Sie einen Trace für alle vom Client an den Server gesendeten Daten nachrichten durchführen möchten, setzen Sie diesen Parameter auf YES. Andernfalls müssen Sie ihn auf NO setzen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung NO verwendet.

### *receive*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Wenn Sie einen Trace für alle Datennachrichten, die der Client vom Server empfängt, durchführen möchten, setzen Sie diesen Parameter auf YES. Andernfalls müssen Sie ihn auf NO setzen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung N0 verwendet.

### Internal\_tracing

Der Teilschlüssel `Internal_tracing` gibt Trace-Optionen für den internen Betrieb des Clients an. Weitere Informationen zum Durchführen von Traces finden Sie im *Communications Server for Linux Diagnostics Guide*. Die Trace-Optionen werden wie folgt angegeben:

*file1* Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Der vollständige Name der Trace-Datei bzw. der ersten Trace-Datei, wenn der Trace in zwei Dateien erfasst wird. (Vergleichen Sie hierzu die Beschreibung des Parameters *file2*.)

Dieser Parameter ist erforderlich, wenn Sie den internen Trace aktivieren möchten. Außerdem müssen Sie den Parameter *trace\_level* setzen.

*file2* Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Der vollständige Pfadname der zweiten Trace-Datei. Dieser Parameter ist optional. Wenn der Trace in einer Datei und nicht in zwei Dateien erfasst werden soll, nehmen Sie diese Zeile nicht auf.

Sind *file1* und *file2* angegeben, erfolgt die Erfassung des Trace in zwei Dateien. Erreicht die erste Datei die vom Parameter *flip\_size* angegebene Größe, wird der Inhalt der zweiten Datei gelöscht und die Erfassung des Trace in der zweiten Datei fortgesetzt. Wenn diese Datei die von *flip\_size* angegebene Größe erreicht, wird der Inhalt der ersten Datei gelöscht und die Erfassung des Trace in der ersten Datei fortgesetzt. Auf diese Weise wird sichergestellt, dass der Trace lange Zeit fortgesetzt werden kann, ohne übermäßig viel Plattenspeicherplatz zu belegen. Der maximal erforderliche Speicherplatz liegt ungefähr bei dem Zweifachen des Parameters *flip\_size*.

*flip\_size*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_DWORD.

Die maximale Größe der Trace-Datei. Wurden zwei Dateinamen angegeben, wechselt der Trace von einer Datei zur nächsten, wenn die aktuelle Datei diese Größe erreicht hat. Wenn nur ein Dateiname angegeben wurde, wird dieser Parameter ignoriert. Die Dateigröße ist nicht begrenzt.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung 1000000 (Bytes) verwendet.

*trace\_level*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_DWORD.

Die Ausführlichkeitsebene des Trace. Gültige Werte reichen von 0 (vollständiger Trace) bis 20 (kein Trace).

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung 20 (kein Trace) verwendet.

*trace\_flushing*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Ist dieser Parameter auf YES gesetzt, wird jede Trace-Anweisung sofort auf Platte ausgelagert. Dies verlangsamt den Betrieb erheblich, stellt jedoch sicher, dass die Trace-Daten bei einem Absturz nicht verloren gehen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung N0 verwendet.

### Appl\_Name

Der Teilschlüssel *Appl\_Name* gibt Optionen für eine CPI-C-Anwendung an.

**Anmerkung:** Diese Optionen können in Umgebungsvariablen oder in der Registrierungsdatenbank angegeben werden. Communications Server für Linux überprüft zuerst die Umgebungsvariablen und verwendet die ggf. darin enthaltenen Informationen. Der Eintrag in der Registrierungsdatenbank wird nur verwendet, wenn die Umgebungsvariable nicht angegeben ist. Die Verwendung von Umgebungsvariablen kann angezeigt sein, wenn Sie Windows Terminal Server verwenden und mehrere Kopien einer Anwendung mit verschiedenen lokalen LUs ausführen müssen.

Wenn Sie diese Optionen für Anwendungen in der Registrierungsdatenbank setzen möchten, nehmen Sie für jede dieser Anwendungen einen Abschnitt dieser Form auf und ersetzen Sie die Variable *Appl\_Name* durch den Namen der ausführbaren Datei des Anwendungsprogramms (ohne jedoch die Dateinamenerweiterung **.exe** aufzunehmen).

Weitere Informationen zu CPI-C finden Sie im *Communications Server for Linux CPI-C Programmer's Guide*.

Die Optionen werden wie folgt angegeben:

#### APPCLLU

Wenn Sie diese Option nicht in der Registrierungsdatenbank angeben möchten, verwenden Sie die Umgebungsvariable APPCLLU.

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Der Name der lokalen LU, die von dieser Anwendung genutzt wird.

Dieser Parameter ist optional. Ist er nicht angegeben, versucht die Anwendung, die Standard-LU (die dem Steuerpunkt des lokalen Knotens zugeordnete LU) zu nutzen.

#### APPCTPN

Wenn Sie diese Option nicht in der Registrierungsdatenbank angeben möchten, verwenden Sie die Umgebungsvariable APPCTPN.

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Der TP-Name der Anwendung. Mit diesem Namen wird die Anwendung in Protokoll- und Trace-Dateien bezeichnet. Bei einer aufgerufenen Anwendung (die *Accept\_Conversation* absetzt) wird diese Option auch für den Abgleich des TP-Namens in einer ankommenden *Allocate*-Anforderung mit der korrekten Anwendung benutzt. Die aufgerufene Anwendung kann mit dem Aufruf *Specify\_Local\_TP\_Name* auch zusätzliche Namen angeben, die mit ankommenden *Allocate*-Anforderungen verglichen werden sollen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird die Standardeinstellung *CPIC\_DEFAULT\_TPNAME* verwendet.

### CSV\_data

Der Teilschlüssel *CSV\_data* gibt Optionen für Anwendungen an, die die Schnittstelle CSV verwenden. Er gilt nur für Anwendungen, die für Zeichenkonvertierungen mit einer benutzerdefinierten Konvertierungstabelle (Tabelle G) das Verb *CONVERT* verwenden. Weitere Informationen zum Verb *CONVERT* finden Sie im *Communications Server for Linux CSV Programmer's Guide*.

Wird diese Funktion von keiner Anwendung auf dem Client genutzt, müssen Sie diesen Abschnitt nicht aufnehmen.

In diesem Abschnitt kann nur die folgende Option angegeben werden:

### *CSVTLG*

Dieser Wert hat in der Registrierungsdatenbank den Datentyp REG\_SZ.

Der vollständige Pfadname der Datei mit der benutzerdefinierten Konvertierungstabelle (Tabelle G). Dieser Parameter ist erforderlich, wenn CSV-Anwendungen eine Zeichenkonvertierung mit Tabelle G durchführen müssen. (Es gibt keine Standardeinstellung.) Andernfalls ist der Parameter optional.



---

## Remote API Clients unter AIX oder Linux verwalten

UNIX

Der Remote API Client kann unter AIX, Linux, Linux für pSeries oder Linux für System z ausgeführt werden.

Die Clientdaten für einen Remote API Client unter AIX oder Linux werden in der Datei **sna\_clnt.net** gespeichert. Diese Datei wird erstellt, wenn Sie die SNA-Software auf dem Client installieren. Diese Datei muss vorhanden sein, damit Sie die Clientsoftware aktivieren können.

**Anmerkung:** Falls der Client für den Serverzugriff HTTPS verwendet, müssen Sie die Datei **sna\_clnt.net** ändern. Geben Sie die Namen der Server an und den WebSphere-Server, der den HTTPS-Zugriff ermöglicht, bevor Sie den Client verwenden. Weitere Informationen hierzu finden Sie im Abschnitt „Netzdatendatei des Clients (sna\_clnt.net)“ auf Seite 160.

## Remote API Clients unter AIX oder Linux aktivieren

Geben Sie an der Eingabeaufforderung den folgenden Befehl ein, um die Software Remote API Client unter AIX oder Linux zu aktivieren:

```
sna start [ -t ]
```

Bei der Installation des Clients aktualisiert das Installationsdienstprogramm automatisch die Startdatei **/etc/rc.sna** (AIX) oder **/etc/rc.d/init.d/snastart** (Linux) und fügt den Befehl **sna start** hinzu. Auf diese Weise wird sichergestellt, dass der Client beim Systemstart automatisch gestartet wird. Falls Sie nicht möchten, dass er automatisch gestartet wird, können Sie diese Zeile löschen oder auf Kommentar setzen und dann die in diesem Abschnitt enthaltenen Anweisungen für das manuelle Aktivieren der Software ausführen.

Es kann nur die folgende Option angegeben werden:

**-t** Aktiviert den Client-Server-Trace. Mit diesem Trace können Sie Fehler bestimmen, die auftreten, wenn der Client versucht, eine Verbindung zu einem Server herzustellen. Wenn Sie diese Option nicht verwenden, ist der

## Remote API Clients unter AIX oder Linux verwalten

Client-Server-Trace für alle Schnittstellen inaktiv. Sie können ihn bei Bedarf im Befehlszeilenverwaltungsprogramm **snaadmin** aktivieren.

Diese Option ist äquivalent zur Auswahl des Feldes *Trace aktivieren* im Motif-Verwaltungsprogramm, nur dass mit dieser Option nicht der DLC-Trace aktiviert wird.

Der Trace setzt die Leistung der CS-Linux-Komponenten herab. Nach Aktivierung der Software können Sie den Trace im Befehlszeilenverwaltungsprogramm **snaadmin** stoppen, wenn er nicht mehr erforderlich ist. Weitere Informationen zum Durchführen von Traces finden Sie im *Communications Server for Linux Diagnostics Guide*.

Geben Sie an der Eingabeaufforderung den folgenden Befehl ein, um den Remote API Client zu stoppen:

```
sna stop
```

### Netzdatendatei des Clients (sna\_clnt.net)

Die Datei **sna\_clnt.net** definiert die Funktionen von Communications Server für Linux, die auf einem Remote API Client unter AIX oder Linux verfügbar sind, und die Server, auf die der Client zugreifen kann. (Weitere Hinweise zum Konfigurieren der äquivalenten Datei auf einem Windows-Client finden Sie in Kapitel 10, „Client-Server-Systeme mit Communications Server für Linux verwalten“, auf Seite 137.)

Die Datei enthält auch Informationen zum Konfigurieren der IP-Port-Nummern, die Communications Server für Linux für die Client-Server-Kommunikation verwendet. Die Standard-Port-Nummern sollten für die meisten Situationen geeignet sein. Sie müssen sich nur mit diesen Informationen beschäftigen, wenn Communications Server für Linux Fehlernachrichten protokolliert, die angeben, dass eine Port-Nummer einen Konflikt mit einem anderen Programm auf demselben Computer auslöst.

Ein Clientcomputer enthält keine Kopie der Domänenkonfigurationsdatei oder der SNA-Netzdatendatei. Er enthält nur die Daten, die er für den Zugriff auf Server im CS-Linux-LAN benötigt, und greift wegen der Bereitstellung der erforderlichen Konfigurationsdaten auf einen Server zurück.

Die notwendigen SNA-Netzdaten sind in der Datei **sna\_clnt.net** im Verzeichnis **/etc/sna** (AIX) oder im Verzeichnis **/etc/opt/ibm/sna** (Linux) enthalten. Diese Datei wird während der Clientinstallation angelegt. Es ist eine ASCII-Datei, die später bei Bedarf mit einem Standardtexteditor modifiziert werden kann.

**Anmerkung:** Wenn Sie die Parameter in dieser Datei geändert haben, müssen Sie den Client stoppen und neu starten, um die Änderungen in Kraft zu setzen. Ausführliche Informationen zur Ausführung dieses Schrittes finden Sie im Abschnitt „Remote API Clients unter AIX oder Linux aktivieren“ auf Seite 159.

Die Datei hat folgenden Inhalt:

```
domain = Domänenname
maximum_process_count = nn
maximum_header_count = nn
maximum_element_count = nn
invoked_tps = YES | NO
lan_access_timeout = nn
```

```
broadcast_attempt_count = nn
server_lost_timeout = nn
*
[ Webservername : [ Port-Nummer : ] ]Servername1
[ Webservername : [ Port-Nummer : ] ]Servername2
.
.
.
```

Nachfolgend sind Beschreibungen der Parameter in den einzelnen Zeilen der Datei aufgelistet:

*domain* Der Parameterwert *Domänenname* gibt den Domännennamen für das CS-Linux-LAN an. Dieser Name wird während der Clientinstallation auf `ibmcs_domain` gesetzt. Diese Zeile ist erforderlich.

*maximum\_process\_count*

Geben Sie die maximale Gesamtanzahl APPC-, CPI-C-, LUA- und NOF-Anwendungen an, die gleichzeitig auf diesem Client ausgeführt werden können.

Dieser Parameter ist optional. Der Standardwert liegt bei 240 und ist normalerweise ausreichend. Falls Sie Fehlermeldungen sehen, die darauf hinweisen, dass ein IPC-Steuerblock nicht zugeordnet werden konnte, müssen Sie möglicherweise die maximale Prozessanzahl erhöhen, indem Sie diesen Parameter angeben. Der größtmögliche Wert ist 4096.

*maximum\_header\_count, maximum\_element\_count*

Diese beiden Parameter sind optional. Die Standardwerte liegen bei 1250 und 1800, was normalerweise ausreichend ist. Sofern Sie nicht von der Benutzerunterstützung dazu aufgefordert werden, müssen Sie in der Regel keine Werte für diese Parameter angeben.

*invoked\_tps*

Geben Sie `invoked_tps = YES` an, wenn auf diesem Client aufgerufene Transaktionsprogramme ausgeführt werden (APPC-TPs, die das Verb `RECEIVE_ALLOCATE` absetzen, oder CPI-C-Anwendungen, die das Verb `Accept_Conversation` bzw. `Accept_Incoming` absetzen). In diesem Fall müssen Sie unter Umständen auch das Transaktionsprogramm auf diesem Client definieren. Weitere Informationen hierzu finden Sie im Abschnitt „TPs definieren“ auf Seite 93.

Geben Sie `invoked_tps = NO` an, wenn der Client nicht für die Ausführung aufgerufener Transaktionsprogramme verwendet wird.

Diese Zeile ist optional. Wenn sie nicht angegeben wird, wird der Standardwert `NO` verwendet.

*lan\_access\_timeout*

Geben Sie die Zeit in Sekunden an, die die IP- oder HTTPS-Verbindung vom Client zu einem Server aktiv bleiben soll, wenn keine Anwendungen auf dem Client CS-Linux-Ressourcen verwenden. Weitere Informationen hierzu finden Sie im Abschnitt „Zeitlimit für LAN-Zugriff“ auf Seite 141.

Das Mindestzeitlimit liegt bei 60 Sekunden (niedrigere Werte werden auf 60 Sekunden hochgesetzt). Wenn Sie die Verbindung schneller beenden möchten, inaktivieren Sie die Software Communications Server für Linux auf dem Client.

Falls Sie kein Zeitlimit angeben möchten, so dass die Verbindung so lange aktiv bleibt, wie die Software Communications Server für Linux auf dem Client ausgeführt wird, geben Sie diesen Parameter nicht an.

## Remote API Clients unter AIX oder Linux verwalten

Dieser Parameter ist optional. Ist er nicht angegeben, wird standardmäßig kein Zeitlimit verwendet.

### *broadcast\_attempt\_count*

Wenn der Client mit der Broadcast-Methode Kontakt zu einem Server aufnimmt (wie vom Parameter \* angegeben), gibt dieser Parameter die maximale Anzahl von Broadcasts für einen Verbindungsversuch zu einem Server an. Der Mindestwert ist 1. Wenn Sie einen höheren Wert angeben, wiederholt der Client seinen Versuch alle 10 Sekunden, bis der Kontakt zu einem Server hergestellt oder dieser Maximalwert erreicht ist. Wird der Maximalwert erreicht, ohne dass eine Verbindung zu einem Server hergestellt wurde, versucht der Client, Kontakt zu einem benannten Server aufzunehmen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird der Standardwert 5 verwendet.

### *server\_lost\_timeout*

Wenn der Client den Kontakt zu einem Server verliert und die Verbindung erneut herstellen muss oder wenn er weder im Broadcast-Betrieb noch unter Verwendung eines benannten Servers Kontakt zu einem Server aufnehmen kann, gibt dieser Parameter die Zeit in Sekunden an, die der Client abwartet, bevor er versucht, eine Verbindung zu einem Server herzustellen. Für den Fall, dass der Client den Kontakt zu einem Server verloren hat, wartet Communications Server für Linux nicht diese ganze Zeitspanne, sondern unternimmt nach einer wahlfreien Zeit zwischen fünf Sekunden und dem angegebenen Zeitlimit einen erneuten Versuch. Auf diese Weise sollen Überlastungen beim Datenaustausch im Netz vermieden werden, die durch eine große Anzahl von Clients ausgelöst werden könnten, wenn diese zur selben Zeit versuchen, Kontakt zu einem Server aufzunehmen.

Dieser Parameter ist optional. Ist er nicht angegeben, wird der Standardwert von 200 Sekunden verwendet.

- \* Diese Zeile gibt an, dass der Client sich in demselben privaten Netz wie seine Server befindet und mit einer UDP-Broadcast-Nachricht an alle Computer in seinem TCP/IP-Teilnetz (oder an alle erreichbaren Teilnetze, wenn der Clientcomputer mehr als eine LAN-Adapterkarte enthält) versuchen soll, Kontakt zu einem Server mit Communications Server für Linux aufzunehmen.

Diese Option ist nur verfügbar, wenn der Client IPV4-Adressen verwendet. UDP-Broadcasts werden für IPV6 nicht unterstützt.

Der Client wiederholt den Broadcast-Versuch alle 10 Sekunden, bis die vom Parameter *broadcast\_attempt\_count* angegebene Anzahl von Versuchen erreicht ist oder der Kontakt zu einem Server hergestellt wurde. Wird die von *broadcast\_attempt\_count* angegebene Begrenzung erreicht, bevor der Kontakt zu einem Server hergestellt werden kann, versucht der Client, direkte Nachrichten an benannte Server (die von den folgenden Zeilen der Datei angegeben werden) zu senden.

Die Verwendung von UDP-Broadcasts ist optional. Wenn keine Broadcast-Versuche unternommen werden sollen, nehmen Sie diese Zeile nicht auf. Wenn diese Zeile vorhanden ist, muss sie vor allen Servernamen in der Datei stehen.

Nehmen Sie diese Zeile nicht auf, wenn der Client sich in demselben privaten Netz wie die Server befindet, jedoch mit UDP-Broadcasts keinen Server erreichen kann. Dies gilt in den folgenden Fällen:

- Das CS-Linux-LAN umfasst mehrere TCP/IP-Teilnetze und nicht alle TCP/IP-Teilnetze, die der Client mit UDP erreichen kann, enthalten CS-Linux-Server.
- Auf dem Client ist keine UDP-Unterstützung installiert.

Wenn der Client mit HTTPS auf die Server zugreift, werden UDP-Broadcasts nicht unterstützt. Geben Sie in diesem Fall die Servernamen explizit wie nachfolgend beschrieben an.

### *Servernamen*

Geben Sie die Namen von CS-Linux-Konfigurationsservern an, zu denen der Client Kontakt aufnehmen soll, um Konfigurationsinformationen zu erhalten. Diese Liste sollte die Namen des Hauptkonfigurationsservers und aller Ausweichserver enthalten, die sich in derselben Domäne wie der Client befinden. Informationen zum Konfigurieren von Haupt- und Ausweichservern finden Sie im Abschnitt „Client-Server-Funktionen konfigurieren“ auf Seite 57.

Falls der Client IPV6-Adressen verwendet, werden keine UDP-Broadcasts unterstützt. Sie müssen an Stelle der Option \* mindestens einen Servernamen angeben.

Wenn der Client mit HTTPS auf die Server zugreift, werden UDP-Broadcasts nicht unterstützt. Geben Sie in diesem Fall den Namen des WebSphere-Servers, der die HTTPS-Unterstützung bereitstellt, und den Namen des CS-Linux-Servers im folgenden Format an:

*Webservername : Servername1*

Hierfür wird vorausgesetzt, dass WebSphere für die Verwendung von HTTPS-Verbindungen am Standard-Port 443 konfiguriert ist. Falls Ihr Netzadministrator WebSphere für die Verwendung eines anderen Ports konfiguriert hat, nehmen Sie die Port-Nummer im folgenden Format auf:

*Webservername : Port-Nummer : Servername1*

Ausführliche Informationen zum Konfigurieren von WebSphere für die Unterstützung von HTTPS-Verbindungen finden Sie in der Veröffentlichung *Einstieg in Communications Server für Linux*.

Wenn die Zeile \* (zur Angabe der Verwendung von UDP-Broadcasts) fehlt oder der Client mit dieser Methode versucht hat, Kontakt zu einem Server aufzunehmen, ohne jedoch eine Antwort zu erhalten, versucht der Client den ersten aufgelisteten Server mit einer direkten Nachricht zu erreichen. Schlägt dieser Versuch fehl, versucht der Client es mit dem zweiten aufgelisteten Server usw. Indem Sie die Reihenfolge ändern, in der die Server aufgelistet sind, können Sie die Last auf mehrere Konfigurationsserver verteilen.

Wenn Sie die Zeile \* (für die Verwendung von UDP-Broadcasts) nicht aufnehmen, müssen Sie mindestens einen Servernamen angeben. Andernfalls sind die Servernamen optional.

Ist der Versuch des Clients, zu jedem der aufgelisteten Server Kontakt aufzunehmen, erfolglos, wartet der Client die vom Parameter *server\_lost\_timeout* angegebene Zeit und versucht dann erneut, eine Verbindung zu einem Server herzustellen (mit UDP-Broadcasts oder dem ersten Server in der Liste).

## Remote API Clients unter AIX oder Linux verwalten

In demselben Verzeichnis (`/etc/sna` unter AIX oder `/etc/opt/ibm/sna` unter Linux) wird neben `sna_clnt.net` die zusätzliche Datei `server.current` gespeichert. Dies ist eine Textdatei, die ggf. den Namen des Servers enthält, mit dem der Client derzeit verbunden ist. Anhand dieser Datei können Sie feststellen, welcher Server für den Client als Verbindungspunkt in die Domäne agiert.



---

## Client-TPs definieren

Informationen zum Definieren von Transaktionsprogrammen auf einem Remote-API-Client-System finden Sie im Abschnitt „TPs definieren“ auf Seite 93 bzw. in Anhang B, „Aufrufbares TP mit Befehlszeilendienstprogramm konfigurieren“, auf Seite 189.

---

## Anhang A. Planungsblätter für die Konfiguration

Dieser Anhang enthält Arbeitsblätter für die Konfiguration bestimmter Funktionen von Communications Server für Linux. In den Arbeitsblättern sind die wesentlichen Konfigurationsparameter zusammengefasst, die für das Aktivieren der einzelnen Funktionen erforderlich sind. Informationen zu erweiterten Konfigurationsparametern finden Sie in den Kapiteln im Hauptteil dieses Handbuchs und in der Veröffentlichung *Communications Server for Linux Administration Command Reference*.

Wenn Sie alle für das Konfigurieren eines Knotens erforderlichen Informationen zusammenstellen wollen, müssen Sie Arbeitsblätter für die folgenden Bereiche vervollständigen:

### **Knotenkonfiguration**

Vervollständigen Sie eines der im Abschnitt „Arbeitsblätter für Knoten“ genannten Arbeitsblätter. Welches Arbeitsblatt für Sie relevant ist, hängt vom Leistungsspektrum des Knotens und von den Kenndaten des Netzes, in dem der Knoten arbeitet, ab.

### **Konnektivitätskonfiguration**

Vervollständigen Sie eines oder mehrere der im Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168 genannten Arbeitsblätter. Welche Arbeitsblätter für Sie relevant sind, hängt von den für die Kommunikation mit den anderen Systemen Ihres Netzes verwendeten Verbindungsprotokollen ab.

### **Konfiguration von Durchgriffsdiensten**

Vervollständigen Sie für Durchgriffsdienste, die vom Knoten unterstützt werden sollen, die im Abschnitt „Arbeitsblätter für Durchgriffsdienste“ auf Seite 177 enthaltenen Arbeitsblätter.

### **Konfiguration der Anwendungsunterstützung**

Vervollständigen Sie eines oder mehrere der im Abschnitt „Arbeitsblätter für Anwendungsunterstützung“ auf Seite 181 genannten Arbeitsblätter. Welche Arbeitsblätter für Sie relevant sind, hängt von der Art der Anwendungen ab, die vom Knoten unterstützt werden sollen.

---

## Arbeitsblätter für Knoten

Vervollständigen Sie eines der folgenden Arbeitsblätter:

- „APPN-Netz-knoten“
- „APPN-Endknoten“ auf Seite 166
- „APPN-Zweig-netzknoten“ auf Seite 167
- „LEN-Knoten“ auf Seite 167

### **APPN-Netz-knoten**

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten ein APPN-Netz-knoten ist (ein Knoten, der Routing-Dienste in einem APPN-Netz bereitstellt).

## Arbeitsblätter für Knoten

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Parameterdialog für Knoten		
<i>APPN-Unterstützung</i>	Netzknoten	
<i>Steuerpunktname</i>	NETZNAME.CP-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A) Für die Verbindung zu einem VTAM-Host muss dieser Name den Einträgen NETID= und CPNAME= der VTAM-PU-Anweisung entsprechen.	
<i>Aliasname für Steuerpunkt</i>	Bis zu acht Zeichen	
<i>Knoten-ID</i>	8 Hexadezimalziffern	
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168.		
Client-Server-Konfiguration: Für einen eigenständigen Knoten nicht erforderlich.		
<i>Konfigurationsserver?</i>	Soll der Knoten die Rolle eines Konfigurationsservers übernehmen und Informationen zu Domänenressourcen im CS-Linux-LAN speichern?	
Anwendungskonfiguration: Siehe Abschnitt „Arbeitsblätter für Anwendungsunterstützung“ auf Seite 181.		

## APPN-Endknoten

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten ein APPN-Endknoten ist (ein Knoten, der dynamische Routing-Daten verwenden, jedoch keine Routing-Dienste für andere Knoten bereitstellen kann).

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Parameterdialog für Knoten		
<i>APPN-Unterstützung</i>	Endknoten	
<i>Steuerpunktname</i>	NETZNAME.CP-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A) Für die Verbindung zu einem VTAM-Host muss dieser Name den Einträgen NETID= und CPNAME= der VTAM-PU-Anweisung entsprechen.	
<i>Aliasname für Steuerpunkt</i>	Bis zu acht Zeichen	
<i>Knoten-ID</i>	8 Hexadezimalziffern	
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168.		
Client-Server-Konfiguration: Für einen eigenständigen Knoten nicht erforderlich.		
<i>Konfigurationsserver?</i>	Soll der Knoten die Rolle eines Konfigurationsservers übernehmen und Informationen zu Domänenressourcen im CS-Linux-LAN speichern?	
Anwendungskonfiguration: Siehe Abschnitt „Arbeitsblätter für Anwendungsunterstützung“ auf Seite 181.		

## APPN-Zweignetznoten

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten ein APPN-Zweignetznoten ist (d. h. ein Knoten, der für die Endknoten in einem Zweignetz Netznotenfunktionen bereitstellt, jedoch aus Sicht des Hauptnetzes selbst ein Endknoten ist).

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Parameterdialog für Knoten		
<i>APPN-Unterstützung</i>	Zweignetznoten	
<i>Steuerpunktname</i>	NETZNAME.CP-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
	Für die Verbindung zu einem VTAM-Host muss dieser Name den Einträgen NETID= und CPNAME= der VTAM-PU-Anweisung entsprechen.	
<i>Aliasname für Steuerpunkt</i>	Bis zu acht Zeichen	
<i>Knoten-ID</i>	8 Hexadezimalziffern	
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168.		
Client-Server-Konfiguration: Für einen eigenständigen Knoten nicht erforderlich.		
<i>Konfigurationsserver?</i>	Soll der Knoten die Rolle eines Konfigurationsservers übernehmen und Informationen zu Domänenressourcen im CS-Linux-LAN speichern?	
Anwendungskonfiguration: Siehe Abschnitt „Arbeitsblätter für Anwendungsunterstützung“ auf Seite 181.		

## LEN-Knoten

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten ein LEN-Knoten ist (ein Knoten ohne Unterstützung für APPN-Funktionen oder ein eigenständiges System, das nur mit einem Host kommuniziert).

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Parameterdialog für Knoten		
<i>APPN-Unterstützung</i>	LEN-Knoten	
<i>Steuerpunktname</i>	NETZNAME.CP-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
	Für die Verbindung zu einem VTAM-Host muss dieser Name den Einträgen NETID= und CPNAME= der VTAM-PU-Anweisung entsprechen.	
<i>Aliasname für Steuerpunkt</i>	Bis zu acht Zeichen	
<i>Knoten-ID</i>	8 Hexadezimalziffern	
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168.		
Client-Server-Konfiguration: Für einen eigenständigen Knoten nicht erforderlich.		

## Arbeitsblätter für Knoten

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>Konfigurationsserver?</i>	Soll der Knoten die Rolle eines Konfigurationsservers übernehmen und Informationen zu Domänenressourcen im CS-Linux-LAN speichern?	
Anwendungskonfiguration: Siehe Abschnitt „Arbeitsblätter für Anwendungsunterstützung“ auf Seite 181.		

## Arbeitsblätter für Konnektivität

Vervollständigen Sie für jedes Verbindungsprotokoll, das für die Kommunikation mit einem anderen Knoten verwendet wird, eines der folgenden Arbeitsblätter. Bei Bedarf können Sie für einen Port mehr als eine Verbindungsstation konfigurieren.

- „SDLC“
- „Token-Ring“ auf Seite 170
- „Ethernet“ auf Seite 172
- „QLLC (X.25)“ auf Seite 174
- „MultiPath Channel“ auf Seite 175
- „Enterprise Extender (HPR/IP)“ auf Seite 176

### SDLC

Vervollständigen Sie dieses Arbeitsblatt, wenn Konnektivität über das SDLC-Verbindungsprotokoll unterstützt werden soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Dialog für SDLC-Port		
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>SDLC-Kartenummer</i>	0 bis <i>Kartenzahl_minus_1</i>	
<i>Port-Nummer</i>	0 bis <i>Anzahl_der_Karten-Ports_minus_1</i>	
<i>Aktiv bei Initialisierung</i>	Bei Bedarf auswählen	
Leitungsangaben		
<i>Typ</i>	Standleitung	
	Abgehende Wählleitung	
	Ankommende Wählleitung	
<i>Funktion der Verbindungsstation</i>	Aushandelbar	
	Primär	
	Primär für Mehrpunktverbindung	
	Sekundär	
	Sekundär, mehrere PUs	
Für ankommende Wählleitungen oder Standleitungen:		

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>Aufrufadresse</i>	Nur für nicht primäre Ports mit ankommender Wählleitung  Auf einem VTAM-Host wird die Aufrufadresse in der VTAM-PU-Definition mit dem Parameter <i>ADDR=</i> konfiguriert.  Bei einer AS/400 ist die Aufrufadresse der Parameter <i>STNADR</i> der Leitungsbeschreibung.	
Dialog für SDLC-Verbindungsstation		
Felder für Verbindungsstationen		
<i>Name</i>	Bis zu acht Zeichen	
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>Aktivierung</i>	Durch Administrator  Beim Initialisieren des Knotens	
<i>LU-Datenverkehr</i>	Bei Anforderung Jeder  Nur unabhängiger  Nur abhängiger	
Unabhängiger LU-Datenverkehr		
<i>Ferner Knoten</i>	<i>NETZNAME.CP-NAME</i> (jeweils 1–8 EBCDIC-Zeichen des Typs A; optional)  Wenn das ferne System ein VTAM-Host ist, entspricht der Netzname (die ersten acht Zeichen des vollständig qualifizierten Namens) dem Parameter <i>NETID</i> der VTAM-Startliste. Die letzten acht Zeichen entsprechen dem Parameter <i>SSCPNAME</i> der VTAM-Startliste.	
<i>Art des fernen Knotens</i>	Feststellen  Netzknoten  End- oder LEN-Knoten	
Abhängiger LU-Datenverkehr		
<i>Funktion des fernen Knotens</i>	Host  Untergeordnet (SNA Gateway)  Untergeordnet (DLUR)	

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>ID des lokalen Knotens</i>	8 Hexadezimalziffern (standardmäßig auf den Knotennamen gesetzt)  In einer VTAM-Konfiguration müssen die ersten drei Stellen mit dem Parameter <i>IDBLK</i> der PU-Definition und die letzten fünf Stellen mit dem Parameter <i>IDNUM</i> übereinstimmen.  Auf einer AS/400 wird die Knoten-ID über den Parameter <i>EXCHID</i> konfiguriert.	
<i>ID des fernen Knotens</i>	8 Hexadezimalziffern (optional)	
<i>Name der untergeordneten PU</i>	1-8 EBCDIC-Zeichen des Typs A	
<i>Name des übergeordneten DLUS</i>	<i>NETZNAME.LU-NAME</i> (jeweils 1-8 EBCDIC-Zeichen des Typs A)	
<b>Informationen für Verbindungsaufnahme</b>		
<i>Aufrufadresse</i>	Bei Ports mit ankommender Wählleitung wird die Adresse nur für den Port konfiguriert.  2 Hexadezimalstellen: <ul style="list-style-type: none"> <li>• C1 für Punkt-zu-Punkt-Verbindungen</li> <li>• 0xFF für primäre Verbindungsstationen mit abgehender Wählleitung (Zieladresse unbekannt)</li> <li>• Eindeutige Adressen für 'Primär für Mehrpunktverbindung'</li> </ul> Auf einem VTAM-Host wird die Aufrufadresse in der VTAM-PU-Definition mit dem Parameter <i>ADDR=</i> konfiguriert.  Bei einer AS/400 ist die Aufrufadresse der Parameter <i>STNADR</i> der Leitungsbeschreibung.	

## Token-Ring

Vervollständigen Sie dieses Arbeitsblatt, wenn Konnektivität über das Token-Ring-Verbindungsprotokoll unterstützt werden soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<b>Dialog für Token-Ring-SAPs</b>		
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>Token-Ring-Kartennummer</i>	0 bis <i>Kartenanzahl_minus_1</i>	
<i>Lokale SAP-Nummer</i>	Hexadezimal (Vielfaches von 4)	
<i>Aktiv bei Initialisierung</i>	Bei Bedarf auswählen	
<i>Auf Verbindungsnetz definieren</i>	Bei Bedarf auswählen	

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>Name des Verbindungsnetzes</i>	NETZNAME.NAME_DES_VERBINDUNGSNETZES (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
Dialog für Token-Ring-Verbindungsstationen		
Felder für Verbindungsstationen		
<i>Name</i>	Bis zu acht Zeichen	
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>Aktivierung</i>	Durch Administrator	
	Beim Initialisieren des Knotens	
	Bei Anforderung	
<i>LU-Datenverkehr</i>	Jeder	
	Nur unabhängiger	
	Nur abhängiger	
Unabhängiger LU-Datenverkehr		
<i>Ferner Knoten</i>	NETZNAME.CP-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A; optional)	
	Wenn das ferne System ein VTAM-Host ist, entspricht der Netzname (die ersten acht Zeichen des vollständig qualifizierten Namens) dem Parameter NETID der VTAM-Startliste. Die letzten acht Zeichen entsprechen dem Parameter SSCPNAME der VTAM-Startliste.	
<i>Art des fernen Knotens</i>	Feststellen	
	End- oder LEN-Knoten	
	Netzknoten	
Abhängiger LU-Datenverkehr		
<i>Funktion des fernen Knotens</i>	Host	
	Untergeordnet (SNA Gateway)	
	Untergeordnet (DLUR)	
<i>ID des lokalen Knotens</i>	8 Hexadezimalziffern (standardmäßig auf den Knotennamen gesetzt)	
	In einer VTAM-Konfiguration müssen die ersten drei Stellen mit dem Parameter IDBLK der PU-Definition und die letzten fünf Stellen mit dem Parameter IDNUM übereinstimmen.	
	Auf einer AS/400 wird die Knoten-ID über den Parameter EXCHID konfiguriert.	
<i>ID des fernen Knotens</i>	8 Hexadezimalziffern (optional)	
<i>Name der untergeordneten PU</i>	1–8 EBCDIC-Zeichen des Typs A	

## Arbeitsblätter für Konnektivität

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>Name des übergeordneten DLUS</i>	NETZNAME.LU-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
Informationen für Verbindungsaufnahme		
<i>MAC-Adresse</i>	Hexadezimalziffern  Ist das ferne Ende dieser Verbindung ein VTAM-Host, entspricht die MAC-Adresse dem Parameter <i>MACADDR</i> der VTAM-Port-Definition.  Wenn Sie eine Verbindung zu einer AS/400 konfigurieren, ist die MAC-Adresse der Parameter <i>ADPTADR</i> der Leitungsbeschreibung.	
<i>SAP-Nummer</i>	Hexadezimal (Vielfaches von 4)  Ist das ferne Ende dieser Verbindung ein VTAM-Host, entspricht die SAP-Nummer dem Parameter <i>SAPADDR</i> der VTAM-PU-Definition.  Wenn Sie eine Verbindung zu einer AS/400 konfigurieren, ist die SAP-Nummer der Parameter <i>SSAP</i> der Leitungsbeschreibung.	

## Ethernet

Vervollständigen Sie dieses Arbeitsblatt, wenn Konnektivität über das Ethernet-Verbindungsprotokoll unterstützt werden soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Dialog für Ethernet-SAPs		
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>Ethernet-Kartennummer</i>	0 bis <i>Kartenanzahl_minus_1</i>	
<i>Lokale SAP-Nummer</i>	Hexadezimal (Vielfaches von 4)	
<i>Aktiv bei Initialisierung</i>	Bei Bedarf auswählen	
<i>Auf Verbindungsnetz definieren</i>	Bei Bedarf auswählen	
<i>Name des Verbindungsnetzes</i>	NETZNAME.NAME_DES_VERBINDUNGSNETZES (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
<i>Ethernet-Typ</i>	Wählen Sie Standard oder 802.3 aus	
Dialog für Ethernet-Verbindungsstationen		
Felder für Verbindungsstationen		
<i>Name</i>	Bis zu acht Zeichen	
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>Aktivierung</i>	Durch Administrator Beim Initialisieren des Knotens	
<i>LU-Datenverkehr</i>	Bei Anforderung Jeder Nur unabhängiger Nur abhängiger	
<b>Unabhängiger LU-Datenverkehr</b>		
<i>Ferner Knoten</i>	NETZNAME.CP-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A; optional)  Wenn das ferne System ein VTAM-Host ist, entspricht der Netzname (die ersten acht Zeichen des vollständig qualifizierten Namens) dem Parameter NETID der VTAM-Startliste. Die letzten acht Zeichen entsprechen dem Parameter SSCPNAME der VTAM-Startliste.	
<i>Art des fernen Knotens</i>	Feststellen  Netzknoten  End- oder LEN-Knoten	
<b>Abhängiger LU-Datenverkehr</b>		
<i>Funktion des fernen Knotens</i>	Host  Untergeordnet (SNA Gateway)  Untergeordnet (DLUR)	
<i>ID des lokalen Knotens</i>	8 Hexadezimalziffern (standardmäßig auf den Knotennamen gesetzt)  In einer VTAM-Konfiguration müssen die ersten drei Stellen mit dem Parameter IDBLK der PU-Definition und die letzten fünf Stellen mit dem Parameter IDNUM übereinstimmen.  Auf einer AS/400 wird die Knoten-ID über den Parameter EXCHID konfiguriert.	
<i>ID des fernen Knotens</i>	8 Hexadezimalziffern (optional)	
<i>Name der untergeordneten PU</i>	1–8 EBCDIC-Zeichen des Typs A	
<i>Name des übergeordneten DLUS</i>	NETZNAME.LU-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
<b>Informationen für Verbindungsaufnahme</b>		

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>MAC-Adresse</i>	Hexadezimalziffern  Ist das ferne Ende dieser Verbindung ein VTAM-Host, entspricht die MAC-Adresse dem Parameter <i>MACADDR</i> der VTAM-Port-Definition.  Wenn Sie eine Verbindung zu einer AS/400 konfigurieren, ist die MAC-Adresse der Parameter <i>ADPTADR</i> der Leitungsbeschreibung.	
<i>SAP-Nummer</i>	Hexadezimal (Vielfaches von 4)  Ist das ferne Ende dieser Verbindung ein VTAM-Host, entspricht die SAP-Nummer dem Parameter <i>SAPADDR</i> der VTAM-PU-Definition.  Wenn Sie eine Verbindung zu einer AS/400 konfigurieren, ist die SAP-Nummer der Parameter <i>SSAP</i> der Leitungsbeschreibung.	

## QLLC (X.25)

Vervollständigen Sie dieses Arbeitsblatt, wenn Konnektivität über das QLLC-Verbindungsprotokoll (X.25) unterstützt werden soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Dialog für QLLC-Ports		
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>X.25-Kartenummer</i>	0 bis <i>Kartenanzahl_minus_1</i>	
<i>Aktiv bei Initialisierung</i>	Bei Bedarf auswählen	
Dialog für QLLC-Verbindungsstation		
Felder für Verbindungsstationen		
<i>Name</i>	Bis zu acht Zeichen	
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>Aktivierung</i>	Durch Administrator  Beim Initialisieren des Knotens	
<i>LU-Datenverkehr</i>	Bei Anforderung Jeder  Nur unabhängiger  Nur abhängiger	
Unabhängiger LU-Datenverkehr		

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>Ferner Knoten</i>	NETZNAME.CP-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A; optional)  Wenn das ferne System ein VTAM-Host ist, entspricht der Netzname (die ersten acht Zeichen des vollständig qualifizierten Namens) dem Parameter NETID der VTAM-Startliste. Die letzten acht Zeichen entsprechen dem Parameter SSCPNAME der VTAM-Startliste.	
<i>Art des fernen Knotens</i>	Feststellen  Netzknoten  End- oder LEN-Knoten	
<b>Abhängiger LU-Datenverkehr</b>		
<i>Funktion des fernen Knotens</i>	Host  Untergeordnet (SNA Gateway)  Untergeordnet (DLUR)	
<i>ID des lokalen Knotens</i>	8 Hexadezimalziffern (standardmäßig auf den Knotennamen gesetzt)  In einer VTAM-Konfiguration müssen die ersten drei Stellen mit dem Parameter IDBLK der PU-Definition und die letzten fünf Stellen mit dem Parameter IDNUM übereinstimmen.  Auf einer AS/400 wird die Knoten-ID über den Parameter EXCHID konfiguriert.	
<i>ID des fernen Knotens</i>	8 Hexadezimalziffern (optional)	
<i>Name der untergeordneten PU</i>	1–8 EBCDIC-Zeichen des Typs A	
<i>Name des übergeordneten DLUS</i>	NETZNAME.LU-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
<b>Informationen für Verbindungsaufnahme</b>		
<i>Ferne X.25-Adresse</i>	Hexadezimalziffern (nur für SVC); 1–4096 (nur für PVC)	

## MultiPath Channel

Vervollständigen Sie dieses Arbeitsblatt, wenn Konnektivität über das MPC-Verbindungsprotokoll unterstützt werden soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<b>Dialog für MPC-Port</b>		
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>Port-Nummer</i>	Muss mit der Einheitennummer der MPC-Einheit übereinstimmen	

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>Aktiv bei Initialisierung</i>	Bei Bedarf auswählen	
Dialog für MPC-Verbindungsstationen		
Felder für Verbindungsstationen		
<i>Name</i>	Bis zu acht Zeichen	
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>Aktivierung</i>	Durch Administrator	
	Beim Initialisieren des Knotens	
	Bei Anforderung	
Unabhängiger LU-Datenverkehr		
<i>Ferner Knoten</i>	NETZNAME.CP-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A; optional)	
	Wenn das ferne System ein VTAM-Host ist, entspricht der Netzname (die ersten acht Zeichen des vollständig qualifizierten Namens) dem Parameter NETID der VTAM-Startliste. Die letzten acht Zeichen entsprechen dem Parameter SSCPNAME der VTAM-Startliste.	
<i>Art des fernen Knotens</i>	Feststellen	
	Netzknoten	
	End- oder LEN-Knoten	

## Enterprise Extender (HPR/IP)

Vervollständigen Sie dieses Arbeitsblatt, wenn Konnektivität über das Enterprise-Extender-Verbindungsprotokoll unterstützt werden soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Dialog für Enterprise-Extender-Ports		
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>Aktiv bei Initialisierung</i>	Bei Bedarf auswählen	
<i>Protokoll</i>	Gibt an, ob Verbindungsstationen an diesem Port IPV4- oder IPV6-Adressen verwenden	
<i>Lokale IP-Schnittstelle</i>	Die für die IP-Verbindung zu verwendende Kennung der lokalen Netzadapterkarte, wenn Sie Zugriff auf mehrere IP-Netze haben. Falls Sie nur Zugriff auf ein IP-Netz haben, können Sie dieses Feld leer lassen.	
<i>Auf Verbindungsnetz definieren</i>	Bei Bedarf auswählen	
<i>Name des Verbindungsnetzes</i>	NETZNAME.NAME_DES_VERBINDUNGSNETZES (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
Dialog für Enterprise-Extender-Verbindungsstationen		
Felder für Verbindungsstationen		

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>Name</i>	Bis zu acht Zeichen	
<i>SNA-Port-Name</i>	Bis zu acht Zeichen	
<i>Aktivierung</i>	Durch Administrator Beim Initialisieren des Knotens Bei Anforderung	
Unabhängiger LU-Datenverkehr		
<i>Ferner Knoten</i>	NETZNAME.CP-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A; optional)  Wenn das ferne System ein VTAM-Host ist, entspricht der Netzname (die ersten acht Zeichen des vollständig qualifizierten Namens) dem Parameter <i>NETID</i> der VTAM-Startliste. Die letzten acht Zeichen entsprechen dem Parameter <i>SSCPNAME</i> der VTAM-Startliste.	
<i>Art des fernen Knotens</i>	Feststellen  End- oder LEN-Knoten  Netzknoten	
Informationen für Verbindungsaufnahme		
<i>Name des fernen IP-Hosts</i>	IPV4-Adresse in Schreibweise mit Trennzeichen (z. B. 193.1.11.100), IPV6-Hexadezimaladresse mit Doppelpunkt (z. B. 2001:0db8:0000:0000:0000:0000:1428:57ab oder 2001:db8::1428:57ab), Name (z. B. newbox.this.co.uk) oder Aliasname (z. B. newbox). Der Parameter <i>Protokoll</i> für den Port bestimmt, ob es sich um eine IPV4- oder IPV6-Adresse handelt.  Wenn Sie einen Namen oder Aliasnamen angeben, muss das Linux-System diesen (mit der lokalen TCP/IP-Konfiguration oder einem Domänennamensserver) in einen vollständig qualifizierten Namen auflösen können.	

## Arbeitsblätter für Durchgriffsdienste

Vervollständigen Sie die Arbeitsblätter für alle in den folgenden Abschnitten beschriebenen Durchgriffsdienste, die vom lokalen Knoten unterstützt werden sollen:

- „DLUR auf dem lokalen Knoten“ auf Seite 178
- „Durchgriff-DLUR für untergeordnete Knoten“ auf Seite 178
- „SNA Gateway“ auf Seite 179
- „TN Server“ auf Seite 179
- „TN Redirector“ auf Seite 180

## DLUR auf dem lokalen Knoten

Vervollständigen Sie dieses Arbeitsblatt, wenn DLUR auf dem lokalen Knoten unterstützt werden soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Knotenkonfiguration: Siehe Abschnitt „Arbeitsblätter für Knoten“ auf Seite 165.		
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168. Konfigurieren Sie für die Unterstützung von DLUR auf dem lokalen Knoten Konnektivität zum APPN-Netz.		
DLUR PU:		
<i>PU-Name</i>	1–8 EBCDIC-Zeichen des Typs A	
<i>DLUS-Name</i>	NETZNAME.LU-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
<i>Name des Ausweich-DLUS</i>	Dieser Parameter ist optional. NETZNAME.LU-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
<i>PU-ID</i>	8 Hexadezimalziffern  In einer VTAM-Konfiguration müssen die ersten drei Stellen mit dem Parameter <i>IDBLK</i> der PU-Definition und die letzten fünf Stellen mit der Einstellung für <i>IDNUM</i> übereinstimmen.  Auf einer AS/400 wird die PU-ID über den Parameter <i>EXCHID</i> konfiguriert.	
<i>Aktiv bei Initialisierung</i>	Bei Bedarf auswählen	
<i>Komprimierung unterstützt</i>	Bei Bedarf auswählen	
<i>Unbegrenzte Verbindungsversuche zum DLUS</i>	Bei Bedarf auswählen	
Konfiguration der lokalen LU und Anwendungskonfiguration: Siehe Abschnitt „Arbeitsblätter für Anwendungsunterstützung“ auf Seite 181. Sie müssen lokale abhängige LUs und die gesamte erforderliche Anwendungsunterstützung konfigurieren.		

## Durchgriff-DLUR für untergeordnete Knoten

Wenn der lokale Knoten ein APPN-Netzknoten ist, können Sie für untergeordnete Knoten Durchgriff-DLUR-Dienste bereitstellen. Vervollständigen Sie dieses Arbeitsblatt, wenn DLUR unterstützt werden soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Knotenkonfiguration: Konfigurieren Sie den Knoten als Netzknoten (siehe Abschnitt „APPN-Netzknoten“ auf Seite 165).		
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168. Konfigurieren Sie Konnektivität zum APPN-Netz und Konnektivität für abhängigen Verkehr zu untergeordneten Knoten.		

## SNA Gateway

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten SNA Gateway unterstützen soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Knotenkonfiguration: Siehe Abschnitt „Arbeitsblätter für Knoten“ auf Seite 165.		
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168. Konfigurieren Sie Konnektivität für abhängigen Verkehr zum Host und Verbindungen für abhängigen Verkehr zu den einzelnen untergeordneten Knoten.		
Konfiguration der lokalen LU und Anwendungskonfiguration: Siehe Abschnitt „Arbeitsblätter für Anwendungsunterstützung“ auf Seite 181.		
Dialog für LU-Pools		
<i>Poolname</i>	1–8 EBCDIC-Zeichen des Typs AE	
<i>LU-Listen</i>	Namen der LUs (der Typen 0–3), die dem Pool zugeordnet werden sollen	
Dialog für untergeordnete LUs		
<i>Name der untergeordneten LU</i>	1–8 EBCDIC-Zeichen des Typs A (1–5 Zeichen für den Basisnamen eines LU-Bereichs)	
<i>Name der untergeordneten PU</i>	EBCDIC-Zeichenfolge des Typs A	
<i>LU-Nummern</i>	1–255 (geben Sie für einen Bereich die Nummer der ersten und der letzten LU an)	
<i>Name der übergeordneten LU</i>	EBCDIC-Zeichenfolge des Typs A (für LU-Name) oder EBCDIC-Zeichenfolge des Typs AE (für Name des LU-Pools)	

## TN Server

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten TN3270-Clients unterstützen soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Knotenkonfiguration: Siehe Abschnitt „Arbeitsblätter für Knoten“ auf Seite 165.		
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168 (konfigurieren Sie Konnektivität für abhängigen LU-Verkehr zum Host).		
Konfiguration der lokalen LU und Anwendungskonfiguration: Siehe Abschnitt „Arbeitsblätter für Anwendungsunterstützung“ auf Seite 181.		
Dialog für LU-Pools		
<i>Poolname</i>	1–8 EBCDIC-Zeichen des Typs AE	
<i>LU-Listen</i>	Namen der LUs (der Typen 0–3), die dem Pool zugeordnet werden sollen	
Dialog für TN-Server-Zugriff		

## Arbeitsblätter für Durchgriffsdienste

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>TN3270-Clientadresse</i>	Geben Sie einen der folgenden Werte an: <ul style="list-style-type: none"> <li>• Standardeintrag (alle TN3270-Clients)</li> <li>• TCP/IP-Adresse (IP-Adresse des Clients im IPV4- oder IPV6-Format)</li> <li>• TCP/IP-Name oder -Aliasname</li> </ul>	
<i>TN3270E unterstützen</i>	Auswählen, wenn (zusätzlich zu TN3270 und TN3287) TN3270E unterstützt werden soll	
<b>TN3270-Port und LUs</b>		
<i>TCP/IP-Port-Nummer</i>	In der Regel 23	
<i>Zugeordnete Anzeige-LU</i>	LU-Name oder Poolname	
<i>Zugeordnete Drucker-LU</i>	LU-Name oder Poolname	
<i>Zugriff auf bestimmte LU erlauben</i>	Bei Bedarf auswählen	
<i>Sichere SSL-Sitzung</i>	Bei Bedarf auswählen	
<i>Clientauthentifizierung ausführen</i>	Bei Bedarf auswählen	
<i>Verschlüsselungsstärke</i>	Geben Sie einen der folgenden Werte an: <ul style="list-style-type: none"> <li>• Nur authentifizieren</li> <li>• Authentifizieren (Minimum)</li> <li>• 40-Bit (Minimum)</li> <li>• 56-Bit (Minimum)</li> <li>• 128-Bit (Minimum)</li> <li>• 168-Bit (Minimum)</li> </ul>	
<b>Dialog für TN-Server-Zuordnung</b>		
<i>Anzeige-LU</i>	LU-Name	
<i>Drucker-LU</i>	LU-Name	

## TN Redirector

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten TN Redirector zur Unterstützung von Telnet-Clients verwenden soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Knotenkonfiguration: Siehe Abschnitt „Arbeitsblätter für Knoten“ auf Seite 165.		
Dialog für TN-Redirector-Zugriff		

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>Adresse des Telnet-Clients</i>	Geben Sie einen der folgenden Werte an: <ul style="list-style-type: none"> <li>• Standardeintrag (alle Telnet-Clients)</li> <li>• TCP/IP-Adresse (IP-Adresse des Clients im IPV4- oder IPV6-Format)</li> <li>• TCP/IP-Name oder -Aliasname</li> </ul>	
<i>TCP/IP-Port-Nummer</i>	In der Regel 23	
<i>Sichere SSL-Sitzung</i>	Bei Bedarf auswählen	
<i>Clientauthentifizierung ausführen</i>	Bei Bedarf auswählen	
<i>Verschlüsselungsstärke</i>	Geben Sie einen der folgenden Werte an: <ul style="list-style-type: none"> <li>• Nur authentifizieren</li> <li>• Authentifizieren (Minimum)</li> <li>• 40-Bit (Minimum)</li> <li>• 56-Bit (Minimum)</li> <li>• 128-Bit (Minimum)</li> <li>• 168-Bit (Minimum)</li> </ul>	
<i>Hostadresse</i>	Geben Sie einen der folgenden Werte an: <ul style="list-style-type: none"> <li>• TCP/IP-Adresse (IP-Adresse des Hosts im IPV4- oder IPV6-Format)</li> <li>• TCP/IP-Name oder -Aliasname</li> </ul>	
<i>TCP/IP-Port-Nummer</i>	Bei Bedarf auswählen	
<i>Sichere SSL-Sitzung</i>	Bei Bedarf auswählen	
<i>Verschlüsselungsstärke</i>	Geben Sie einen der folgenden Werte an: <ul style="list-style-type: none"> <li>• Nur authentifizieren</li> <li>• Authentifizieren (Minimum)</li> <li>• 40-Bit (Minimum)</li> <li>• 56-Bit (Minimum)</li> <li>• 128-Bit (Minimum)</li> <li>• 168-Bit (Minimum)</li> </ul>	

## Arbeitsblätter für Anwendungsunterstützung

Vervollständigen Sie die Arbeitsblätter für die Benutzeranwendungen, die vom lokalen Knoten unterstützt werden sollen.

- „APPC“
- „CPI-C“ auf Seite 185
- „5250“ auf Seite 185
- „3270“ auf Seite 186
- „LUA“ auf Seite 187

### APPC

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten APPC-Anwendungen unterstützen soll.

## Arbeitsblätter für Anwendungsunterstützung

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Knotenkonfiguration: Siehe Abschnitt „Arbeitsblätter für Knoten“ auf Seite 165.		
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168.		
Dialog für lokale LUs: Nicht erforderlich, wenn Sie die Standardsteuerpunkt-LU verwenden können.		
<i>LU-Name</i>	1–8 EBCDIC-Zeichen des Typs A	
<i>LU-Aliasname</i>	Bis zu acht Zeichen	
Parameter für abhängige LU		
<i>Hostverbindungsstation/Name</i>	Name der abhängigen Verbindungsstation zum Host oder der DLUR-PU (muss vor dem Definieren einer LU definiert werden)	
<i>LU-Nummer</i>	1–255	Dieser Wert muss mit dem Parameter <i>LOCADDR</i> in der VTAM/NCP-Anweisung für LU-Ressourcendefinition übereinstimmen.
<i>Teil des Standardpools</i>	Bei Bedarf auswählen (nur für abhängige LU)	
Parameter für lokale LUs		
<i>Sync-Punkt unterstützen</i>	Bei Bedarf auswählen	
<i>Kennwortsubstitution inaktivieren</i>	Bei Bedarf auswählen	
<i>Auf bestimmten SSCP einschränken</i>	Bei Bedarf auswählen (nur für abhängige LU)	
	Die SSCP-ID ist im Feld <i>SSCPID=</i> der VTAM-Startliste definiert.	
Dialog für fernen Knoten: Nur konfigurieren, wenn der lokale Knoten ein LEN-Knoten ist.		
<i>SNA-Netzname des Knotens</i>	<i>NETZNAME.CP-NAME</i> (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
Dialog für Partner LUs: Nur für die Kommunikation mit einem LEN-Knoten erforderlich, um den Aliasnamen einer Partner-LU zu definieren, oder wenn der lokale Knoten ein LEN-Knoten ist.		
<i>Name der Partner-LU</i>	<i>NETZNAME.LU-NAME</i> (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
<i>Aliasname</i>	Bis zu acht Zeichen	
<i>Nicht interpretierter Name</i>	1–8 EBCDIC-Zeichen des Typs AE (wenn sich der Name der Host-LU von dem lokal verwendeten Namen der Partner-LU unterscheidet)	
<i>Unterstützt Parallelsitzungen</i>	Auswählen, wenn solche Sitzungen unterstützt werden	
<i>Position</i>	<i>NETZNAME.CP-NAME</i> (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
Dialog für Verbindungsstations-Routing: Nur erforderlich, wenn die Partner-LU von der Verbindungsstation lokalisiert wird.		

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>LU-Name</i>	1–8 EBCDIC-Zeichen des Typs A	
<i>Name der Verbindungsstation</i>	Bis zu acht Zeichen	
<i>Name der Partner-LU</i>	NETZNAME.LU-NAME (jeweils 1–8 EBCDIC-Zeichen des Typs A)	
<i>Namen der Partner-LU generisch verwenden</i>	Bei Bedarf auswählen	
Modusdialog: Nur erforderlich, wenn Sie einen Modus verwenden, der kein Standardmodus ist.		
<i>Name</i>	1–8 EBCDIC-Zeichen des Typs A	
<i>Serviceklassenname</i>	1–8 EBCDIC-Zeichen des Typs A	
<b>Sitzungsbegrenzungen</b>		
<i>Einleitende Sitzungsbegrenzung</i>	Bis zur maximalen Sitzungsbegrenzung; empfohlen wird der Wert 8.	
<i>Maximale Sitzungsbegrenzung</i>	Bis zu 32767	
<i>Mindestanzahl Sitzungen für Konfliktgewinner</i>	Bis zur maximalen Sitzungsbegrenzung; empfohlen wird der Wert 0.	
<i>Mindestanzahl Sitzungen für Konfliktverlierer</i>	Der empfohlene Wert ist 0.	
<i>Automatisch aktivierte Sitzungen</i>	0 bis <i>Mindestanzahl_Sitzungen_für_Konfliktgewinner</i>	
<b>Fenster für Empfangsnachrichtendosierung</b>		
<i>Anfangswert</i>	Der empfohlene Wert ist 4.	
<i>Maximum</i>	Optional	
<i>Sitzungszeitlimit</i>		
<i>Maximale RU-Größe</i>	Die empfohlene obere Begrenzung ist 1024.	
<b>Komprimierung unterstützt</b>		
<i>Max. Komprimierung für ankommende Daten</i>	Keine RLE LZ9 LZ10	
<i>Max. Komprimierung für abgehende Daten</i>	Keine RLE LZ9 LZ10	
Dialog für Sitzungssicherheit: Nur erforderlich, wenn für Sitzungen zwischen einer bestimmten lokalen LU und einer Partner-LU Sitzungssicherheit notwendig ist.		
<i>Lokale LU</i>	1–8 EBCDIC-Zeichen des Typs A	
<i>Partner-LU</i>	1–8 EBCDIC-Zeichen des Typs A	
<i>Kennwort</i>	16-stellige Hexadezimalzahl	

## Arbeitsblätter für Anwendungsunterstützung

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Dialog für das Aufrufen von TPs: Nur erforderlich, wenn das lokale Transaktionsprogramm als Reaktion auf Anforderungen von fernen Systemen gestartet werden soll.		
<i>Name des Transaktionsprogramms</i>	Benutzeranwendung: bis zu 64 ASCII-Zeichen  Service-TP: bis zu acht Hexadezimalziffern	
<i>Auf bestimmte LU einschränken</i>	Bei Bedarf auswählen	
<i>LU-Aliasname</i>	Bis zu acht Zeichen	
<i>Ankommende Zuordnungen in Warteschlange stellen</i>	Für TPs ohne Warteschlangenbetrieb auswählen. Wird dieser Wert nicht ausgewählt, werden ankommende Allocate-Anforderungen in eine Warteschlange gestellt, sofern das TP bereits aktiv ist.	
<i>Ankommende Allocate-Anforderungen an aktives TP weiterleiten</i>	Für TPs mit Broadcast- und Warteschlangenbetrieb auswählen	
<i>Vollständiger Pfad zur ausführbaren Datei des Transaktionsprogramms</i>	Pfad- und Dateiname der ausführbaren Datei (standardmäßig <i>Name des Transaktionsprogramms</i> )	
<i>Argumente</i>	Alle für die ausführbare Datei gültigen Argumente	
<i>Benutzer-ID</i>	Bis zu 64 Zeichen	
<i>Gruppen-ID</i>	Bis zu 64 Zeichen	
Definitionsdialog für TPs: Definiert APPC-Kenndaten.		
<i>Name des Transaktionsprogramms</i>	Benutzeranwendung: bis zu 64 ASCII-Zeichen  Service-TP: bis zu acht Hexadezimalziffern	
<i>Sicherheit auf Dialogebene erforderlich</i>	Auswählen, wenn bei Zuordnungsanforderungen ein gültiger Benutzername und ein gültiges Kennwort erforderlich sein sollen	
<i>Zugriff einschränken</i>	Auswählen, wenn Benutzernamen in eine Sicherheitszugriffsliste aufgenommen werden sollen	
<i>Sicherheitszugriffsliste Dialogart</i>	Name der Sicherheitszugriffsliste Formatiert  Formatfrei	
<i>Synchronisationsebene</i>	Beides Keine  Bestätigen  Synchronisationspunkt  Keine oder Bestätigen  Keine, Bestätigen oder Synchronisationspunkt	

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
<i>PIP erlaubt</i>	Bei Bedarf auswählen	
Dialog für Dialogsicherheit: Nur erforderlich, wenn für ein lokales Transaktionsprogramm, das als Reaktion auf Anforderungen von fernen Systemen gestartet werden soll, Dialogsicherheit benötigt wird.		
<i>Benutzer-ID</i>	Bis zu zehn Zeichen	
<i>Kennwort</i>	Bis zu zehn Zeichen	

### CPI-C

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten CPI-C-Anwendungen unterstützen soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Knotenkonfiguration: Siehe Abschnitt „Arbeitsblätter für Knoten“ auf Seite 165.		
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168.		
APPC-Konfiguration: Siehe Abschnitt „APPC“ auf Seite 181.		
Dialog für CPI-C-Bestimmungsorte		
<i>Symbolischer Bestimmungsort</i>	1–8 Zeichen	
<i>Lokale LU</i>	Aliasname (bis zu acht Zeichen) oder vollständig qualifizierter Name ( <i>NETZNAME.LU-NAME</i> mit jeweils 1–8 EBCDIC-Zeichen des Typs A)	
<i>Partner-LU</i>	Aliasname (bis zu acht Zeichen) oder vollständig qualifizierter Name ( <i>NETZNAME.LU-NAME</i> mit jeweils 1–8 EBCDIC-Zeichen des Typs A)	
<i>Modus</i>	EBCDIC-Zeichenfolge des Typs A	
<i>Name des Partner-TP</i>	Benutzeranwendung: bis zu 64 Zeichen	
	Service-TP: bis zu acht Hexadezimalziffern	
<i>Sicherheit</i>	Keine	
	Dieselbe	
<i>Benutzer-ID</i>	Programm Nur für die Sicherheitsstufen Dieselbe und Programm (kein Bezug zur Anmelde-ID eines Benutzers)	
<i>Kennwort</i>	Nur für die Sicherheitsstufe Programm (kein Bezug zum Anmeldekennwort eines Benutzers)	

### 5250

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten die 5250-Kommunikation unterstützen soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
	Knotenkonfiguration: Siehe Abschnitt „Arbeitsblätter für Knoten“ auf Seite 165.	
	Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168 (konfigurieren Sie Konnektivität für unabhängigen Verkehr).	
	APPC-Konfiguration: Siehe Abschnitt „APPC“ auf Seite 181.	

## 3270

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten die 3270-Kommunikation unterstützen soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
	Knotenkonfiguration: Siehe Abschnitt „Arbeitsblätter für Knoten“ auf Seite 165.	
	Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168 (konfigurieren Sie Konnektivität für abhängigen Verkehr).	
	Dialog für LUs der Typen 0–3	
<i>LU-Name</i>	1–8 EBCDIC-Zeichen des Typs A (oder 1–5 Zeichen für den Basisnamen eines LU-Bereichs)	
<i>Hostverbindungsstation/DLUR-PU</i>	Name der abhängigen Verbindungsstation zum Host oder der DLUR-PU (muss vor dem Definieren einer LU definiert werden)	
<i>LU-Nummern</i>	1–255 (geben Sie für einen Bereich die Nummer der ersten und der letzten LU an)	
<i>LU-Typ</i>	Dieser Wert muss mit dem Parameter <i>LOCADDR</i> in der VTAM/NCP-Anweisung für LU-Ressourcendefinition übereinstimmen.	
	3270 Modell 2 (80x24) Anzeige	
	3270 Modell 3 (80x32) Anzeige	
	3270 Modell 4 (80x43) Anzeige	
	3270 Modell 5 (132x27) Anzeige	
	3270-Drucker	
	SCS-Drucker	
<i>LU im Pool</i>	Gewünschte Option auswählen (nur für Anzeige-LUs und unbeschränkte LUs).	
<i>Poolname</i>	1–8 EBCDIC-Zeichen des Typs AE	
	Dialog für LU-Pools	
<i>Poolname</i>	1–8 EBCDIC-Zeichen des Typs AE	
<i>LU-Listen</i>	Namen der LUs (der Typen 0–3), die dem Pool zugeordnet werden sollen	

## LUA

Vervollständigen Sie dieses Arbeitsblatt, wenn der lokale Knoten LUA-Anwendungen unterstützen soll.

Motif-Feld	Gültige Eingabe / Anmerkungen	Wert Ihrer Implementierung
Knotenkonfiguration: Siehe Abschnitt „Arbeitsblätter für Knoten“ auf Seite 165.		
Konnektivitätskonfiguration: Siehe Abschnitt „Arbeitsblätter für Konnektivität“ auf Seite 168 (konfigurieren Sie Konnektivität für abhängigen Verkehr).		
Dialog für LUs der Typen 0–3		
<i>LU-Name</i>	1–8 EBCDIC-Zeichen des Typs A (oder 1–5 Zeichen für den Basisnamen eines LU-Bereichs)	
<i>Hostverbindungsstation/DLUR-PU</i>	Name der abhängigen Verbindungsstation zum Host oder der DLUR-PU (muss vor dem Definieren einer LU definiert werden)	
<i>LU-Nummern</i>	1–255 (geben Sie für einen Bereich die Nummer der ersten und der letzten LU an)	
	Dieser Wert muss mit dem Parameter <i>LOCADDR</i> in der VTAM/NCP-Anweisung für LU-Ressourcendefinition übereinstimmen.	
<i>LU-Typ</i>	Unbeschränkt	
<i>LU im Pool</i>	Gewünschte Option auswählen (nur für Anzeige-LUs und unbeschränkte LUs).	
<i>Poolname</i>	1–8 EBCDIC-Zeichen des Typs AE	
Dialog für LU-Pools		
<i>Poolname</i>	1–8 EBCDIC-Zeichen des Typs AE	
<i>LU-Listen</i>	Namen der LUs (der Typen 0–3), die dem Pool zugeordnet werden sollen	



---

## Anhang B. Aufrufbares TP mit Befehlszeilendienstprogramm konfigurieren

Zum Lieferumfang von Communications Server für Linux gehört ein Befehlszeilendienstprogramm, mit dem der Benutzer oder Entwickler eines TP-Installationsprogramms ein aufrufbares TP definieren kann. Sie können dieses Dienstprogramm auf einem Server oder Client ausführen.

Die Syntax des Befehls ist je nachdem, ob Sie TP-Definitionen abfragen, entfernen oder definieren, unterschiedlich und für einen Remote API Client unter Windows eine andere:

### Aufrufbares Transaktionsprogramm definieren:

UNIX

**snatpinstall -a** *Dateiname*

WINDOWS

**tpinst32 -a** *Dateiname*

**Anmerkung:** Der Befehl **tpinst32** gilt für 32-Bit- und x64-Versionen von Windows.



Dieser Befehl fügt eine oder mehrere TP-Definitionen aus der angegebenen Datei *Dateiname* hinzu. Wenn das in der Datei angegebene TP bereits definiert wurde, wird die vorhandene Definition durch die Informationen in der Datei überschrieben. Informationen zum erforderlichen Dateiformat finden Sie im Abschnitt „Format der Definitionsdatei für aufrufbares TP“ auf Seite 190.

### Definition eines aufrufbaren Transaktionsprogramms entfernen:

UNIX

**snatpinstall -r -t** *TP-Name* [ **-l** *LU-Aliasname* ]

Dieser Befehl entfernt den Eintrag mit dem angegebenen TP-Namen und (falls mehr als ein APPC-TP mit demselben Namen definiert ist) mit dem angegebenen LU-Aliasnamen. Lassen Sie die Option **-l** *LU-Aliasname* weg, wenn der Eintrag für eine CPI-C-Anwendung bestimmt oder nur ein APPC-TP mit dem angegebenen TP-Namen definiert ist.

## Aufrufbares TP in der Befehlszeilendienstprogramm konfigurieren

WINDOWS

```
tpinst32 -r -t TP-Name
```

Dieser Befehl entfernt den Eintrag mit dem angegebenen TP-Namen.



### Definitionen aufrufbarer Transaktionsprogramme abfragen:

UNIX

```
snatpinstall -q [ -t TP-Name ] [ -l LU-Aliasname ]
```

Dieser Befehl fragt den Eintrag mit dem angegebenen TP-Namen und (falls mehr als ein APPC-TP mit demselben Namen definiert ist) mit dem angegebenen LU-Aliasnamen ab. Lassen Sie die Option **-l** *LU-Aliasname* weg, wenn der Eintrag für eine CPI-C-Anwendung bestimmt oder nur ein APPC-TP mit dem angegebenen TP-Namen definiert ist. Wenn Sie die Option **-t** *TP-Name* nicht angeben, fragt der Befehl alle Definitionen aufrufbarer TPs ab.

WINDOWS

```
tpinst32 -q -t TP-Name
```

Dieser Befehl fragt den Eintrag mit dem angegebenen TP-Namen ab. Wenn Sie die Option **-t** *TP-Name* nicht angeben, fragt der Befehl alle Definitionen aufrufbarer TPs ab.



---

## Format der Definitionsdatei für aufrufbares TP

Die Datei mit den Konfigurationsdaten eines aufrufbaren Transaktionsprogramms ist eine ASCII-Textdatei, die mit einem Standardtexteditor geändert werden kann. Die Einträge der Datei haben das folgende Format:

```
[TP-Name]  
PATH           = vollständiger_Pfadname_der_ausführbaren_Datei  
ARGUMENTS     = durch_Leerzeichen_getrennte_Befehlszeilenargumente  
TYPE          = QUEUED | QUEUED-BROADCAST | NON-QUEUED  
TIMEOUT       = nnn
```

UNIX

```
USERID        = Benutzer-ID  
GROUP         = Gruppen-ID  
LUALIAS       = LU-Aliasname  
ENV           = Umgebungsvariable=Wert  
.  
ENV           = Umgebungsvariable=Wert
```

### WINDOWS

SHOW = MAXIMIZED | MINIMIZED | HIDDEN | NORMAL | NOACTIVATE | MINNOACTIVATE  
SECURITY\_TYPE = APPLICATION | SERVICE  
SERVICE\_NAME = *Name\_des\_installierten\_Dienstes*  
USERID = *Domänenname\Benutzer-ID*



Die Parameter werden wie folgt verwendet. Für ein vom Bediener gestartetes Transaktionsprogramm werden nur der TP-Name, der TP-Typ, das Zeitlimit und (für ein APPC-TP unter AIX oder Linux) der LU-Aliasname verwendet. Die übrigen Parameter gelten nur für automatisch gestartete Transaktionsprogramme.

### UNIX

Unter AIX oder Linux gibt Communications Server für Linux eine Fehlermeldung zurück, wenn Sie einen ungültigen Parameter angeben.

### WINDOWS

Auf Windows-Maschinen ignoriert Communications Server für Linux ungültige Parameter.



#### *TP-Name*

Der Name des TP (1–64 Zeichen ohne eingebettete Leerzeichen). Der im Aufruf `Receive_Allocate` oder in der ankommenden `Allocate`-Anforderung angegebene TP-Name wird mit diesem Namen verglichen. Wenn das TP ein automatisch gestartetes Transaktionsprogramm ist, müssen Sie diesen Namen beim Initialisieren des Verbs `Receive_Allocate` angeben, damit Communications Server für Linux die ankommende `Attach`-Anforderung an das richtige TP weiterleiten kann.

Dieser Name muss in eckige Klammern eingeschlossen werden. Der Name kann als eine in Anführungszeichen gesetzte ASCII-Zeichenfolge angegeben werden (z. B. ["TP-NAME1"]). Alternativ dazu kann der Name auch im Hexadezimalformat mit den EBCDIC-Zeichen des TP-Namens (z. B. [<53504E414D45F1>]) oder als Kombination beider Formate (z. B. [<3f>"TP1"]) angegeben werden. In diesem Beispiel ist das erste Zeichen das nicht druckbare Zeichen 0x3f, und die übrigen Zeichen sind "TP1".

Communications Server für Linux setzt ASCII-Zeichenfolgen in EBCDIC-Zeichen um. Hexadezimalzeichen werden jedoch nicht umgesetzt (da CS Linux voraussetzt, dass diese Zeichen bereits EBCDIC-Zeichen sind). Anschließend wird der Name mit EBCDIC-Leerzeichen aufgefüllt (bis er 64 Zeichen lang ist). Erst dann erfolgt der Abgleich mit dem angegebenen TP-Namen.

## Format der Definitionsdatei für aufrufbares TP

**PATH** Der Pfad- und Dateiname der ausführbaren Datei dieses Transaktionsprogramms.

Diese Zeile ist optional. Wird sie nicht angegeben, geht Communications Server für Linux davon aus, dass der Name der ausführbaren Datei mit dem TP-Namen übereinstimmt. Wenn Sie einen Dateinamen ohne Pfad angeben, wird für AIX- oder Linux-Systeme der Standardpfad **/etc/opt/ibm/sna** verwendet. Bei einem Windows-Client verwendet Communications Server für Linux die normalen Windows-Mechanismen, um die ausführbare Datei zu finden.

### ARGUMENTS

Alle an das Transaktionsprogramm zu übergebenden und durch Leerzeichen voneinander getrennten Argumente. Diese Argumente werden in der Reihenfolge, in der sie in der Befehlszeile erscheinen, an das Transaktionsprogramm übergeben.

Diese Zeile ist optional. Wenn sie nicht angegeben wird, wird das Transaktionsprogramm ohne Befehlszeilenargumente aufgerufen.

**TYPE** Geben Sie einen der folgenden Werte an:

**QUEUED** Das TP ist ein Transaktionsprogramm mit Warteschlangenbetrieb. Alle Allocate-Anforderungen, die ankommen, während das Transaktionsprogramm läuft, werden in eine Warteschlange gestellt, bis das Transaktionsprogramm einen weiteren Aufruf Receive\_Allocate absetzt oder beendet wird und neu gestartet werden kann. Eine ankommende Allocate-Anforderung wird nur dann an dieses Transaktionsprogramm weitergeleitet, wenn sie von einer LU empfangen wird, die für das Routing von Allocate-Anforderungen an diesen Computer konfiguriert ist.

### QUEUED-BROADCAST

Das TP ist ein Transaktionsprogramm mit Broadcast- und Warteschlangenbetrieb. Alle Allocate-Anforderungen, die ankommen, während das Transaktionsprogramm läuft, werden in eine Warteschlange gestellt, bis das Transaktionsprogramm einen weiteren Aufruf Receive\_Allocate absetzt oder beendet wird und neu gestartet werden kann. Nach dem Start des TP werden Informationen zum TP per Broadcast an alle Server im LAN gesendet. Wenn eine LU auf einem anderen Computer, für die keine Routing-Informationen konfiguriert sind, eine ankommende Allocate-Anforderung empfängt, kann sie das TP dynamisch lokalisieren und die Allocate-Anforderung an das TP weiterleiten.

Wenn Sie an Stelle von QUEUED den Wert QUEUED-BROADCAST verwenden, müssen keine expliziten Routing-Informationen für LUs konfiguriert werden. Außerdem kann die Arbeitslast durch Ausführung mehrerer Kopien eines TP auf verschiedenen Computern verteilt werden. Falls Sie jedoch die Informationsübermittlung per Broadcast verhindern möchten, um den Datenverkehr im LAN zu reduzieren, oder falls Sie sicherstellen müssen, dass bei einer bestimmten LU ankommende Allocate-Anforderungen immer an dieselbe Kopie des TP weitergeleitet werden, sollten Sie QUEUED verwenden.

### NON-QUEUED

Das TP ist ein Transaktionsprogramm ohne Warteschlangenbetrieb. Communications Server für Linux startet bei jeder für das TP ankommenden Allocate-Anforderung eine neue Kopie des Transaktionsprogramms. Geben Sie für TPs ohne Warteschlangenbetrieb nicht den Parameter *TIMEOUT* an.

Ein als TP ohne Warteschlangenbetrieb definiertes Transaktionsprogramm kann nicht von einem Bediener gestartet werden. Es wird immer automatisch von Communications Server für Linux gestartet. Soll das TP von einem Bediener gestartet werden, dürfen Sie nicht *NON-QUEUED* angeben. Versucht ein Benutzer, ein Transaktionsprogramm ohne Warteschlangenbetrieb zu starten, weist Communications Server für Linux das Verb *Receive\_Allocate* zurück, da es von keiner ankommenden Allocate-Anforderung erwartet wird.

Wenn ein TP ohne Warteschlangenbetrieb einen Dialog beendet hat, kann es beendet werden oder ein weiteres Verb *Receive\_Allocate* absetzen. Für häufig verwendete Programme bietet dies eine Möglichkeit, den zum Starten einer neuen Programminstanz für jeden Dialog erforderlichen Systemaufwand zu vermeiden. Jedes Mal, wenn eine Attach-Anforderung für ein automatisch gestartetes TP ohne Warteschlangenbetrieb empfangen wird, überprüft Communications Server für Linux, ob bereits ein *Receive\_Allocate* von einer Instanz dieses TP ansteht. Ist dies der Fall, wird das TP für den ankommenden Dialog verwendet. Andernfalls startet Communications Server für Linux eine neue Instanz des Programms.

Bei Verwendung von *NON-QUEUED* können mehrere Kopien des TP gleichzeitig ausgeführt werden. Sollte das Transaktionsprogramm in Dateien schreiben, müssen Sie sicherstellen, dass verschiedene Kopien des Transaktionsprogramms nicht gegenseitig ihre Dateien überschreiben. Verwenden Sie dazu eine der folgenden Methoden:

- Gewährleisten Sie, dass das Transaktionsprogramm keine Datei erstellt, sondern Daten an eine vorhandene Datei anhängt (so dass alle Kopien des Transaktionsprogramms Daten an dieselbe Datei anhängen).
- Stellen Sie das Transaktionsprogramm so ein, dass es zur Laufzeit ausgehend von der Prozess-ID, mit der das TP ausgeführt wird, Dateinamen generiert (so dass jede Kopie des Transaktionsprogramms in eine andere Datei schreibt).

Diese Zeile ist optional. Wenn sie nicht oder mit einem ungültigen Wert angegeben wird, wird der Standardwert *QUEUED* verwendet.

### *TIMEOUT*

Die maximale Zeit in Sekunden, die ein vom TP abgesetzter Aufruf *Receive\_Allocate* blockiert werden soll, wenn keine ankommende Allocate-Anforderung ansteht. Wird in dieser Zeit keine ankommende Allocate-Anforderung empfangen, schlägt der Aufruf mit einem Rückkehrcode, der „Statusprüfung - keine anstehende Allocate-Anforderung“ angibt, fehl.

Ein Zeitlimit von 0 bedeutet, dass der Aufruf immer fehlschlägt, sofern beim Absetzen des Aufrufs nicht bereits eine ankommende Allocate-Anforderung ansteht. Ein Zeitlimit von -1 zeigt an, dass der Aufruf unbegrenzte Zeit auf eine ankommende Allocate-Anforderung wartet.

## Format der Definitionsdatei für aufrufbares TP

Diese Zeile ist optional. Wenn sie nicht oder mit einem ungültigen (einem nicht numerischen) Wert angegeben wird, wird der Standardwert -1 (unendlich) verwendet.

Geben Sie diesen Parameter nicht an, wenn der Parameter *TYPE* auf *NON-QUEUED* gesetzt ist. Communications Server für Linux verwendet das Zeitlimit 0 für Transaktionsprogramme ohne Warteschlangenbetrieb, denn diese werden in Reaktion auf eine ankommende Allocate-Anforderung gestartet, so dass immer eine Anforderung ansteht.

UNIX

### *USERID*

Geben Sie die Benutzer-ID an, die Communications Server für Linux für das Starten des Transaktionsprogramms verwendet. Das Transaktionsprogramm wird in dem dieser Benutzer-ID zugeordneten Benutzerverzeichnis gestartet. Dieses Benutzerverzeichnis ist außerdem der Standardpfad für Trace-Dateien und andere Dateien, auf die das Transaktionsprogramm zugreift (sofern er nicht von der Anwendung mit einem vollständigen Pfad überschrieben wird). Wenn die Anwendung einen Dateinamen ohne Pfad angibt, sucht Communications Server für Linux in diesem Benutzerverzeichnis nach der Datei. Gibt die Anwendung einen Dateinamen mit einem relativen Pfad an, sucht Communications Server für Linux ausgehend von diesem Benutzerverzeichnis im angegebenen relativen Verzeichnis nach der Datei.

Diese Zeile ist erforderlich und muss angegeben werden. Die ID muss eine gültige Anmelde-ID auf dem CS-Linux-Computer sein. Sie kann bis zu 64 Zeichen lang sein, sofern Ihre AIX- oder Linux-Konfiguration keine Einschränkung von Benutzernamen auf weniger Zeichen festlegt.

Der angegebene Benutzer muss für die vom Parameter *PATH* angegebene ausführbare Datei die Ausführungsberechtigung haben. Ist der Parameter *USERID* auf root gesetzt, muss root der Eigner der Datei sein und über die Berechtigungen *setuid* und *setgid* verfügen, damit die Datei von Communications Server für Linux automatisch gestartet werden kann.

### *GROUP*

Geben Sie die Gruppen-ID an, die Communications Server für Linux für das Starten des Transaktionsprogramms verwendet. Die ID muss eine gültige Gruppen-ID auf dem CS-Linux-Computer sein. Sie kann bis zu 64 Zeichen lang sein, sofern Ihre AIX- oder Linux-Konfiguration keine Einschränkung von Gruppennamen auf weniger Zeichen festlegt.

Diese Zeile ist optional. Wenn sie nicht angegeben wird, wird der Standardwert *other* verwendet.

### *LUALIAS*

Geben Sie den Aliasnamen der lokalen LU an, von der das Transaktionsprogramm ankommende Attach-Anforderungen akzeptieren soll.

**Anmerkung:** Dieser Parameter kann nur verwendet werden, wenn das Transaktionsprogramm ein APPC-TP ist. Geben Sie diesen Parameter nicht an, wenn das TP eine CPI-C-Anwendung ist. CPI-C kann keine ankommenden Attach-Anforderungen von einer bestimmten LU akzeptieren. Wenn für eine CPI-C-Anwendung ein LU-Aliasname (auch ein aus Leerzeichen bestehender LU-Name) angegeben wird, führt dies beim Routing der ankommenden Attach-Anforderung an das TP zu Fehlern.

Dieser Name besteht aus acht Zeichen und muss mit dem Namen einer lokalen APPC-LU von Communications Server für Linux übereinstimmen.

Wenn Sie angeben wollen, dass das TP von allen lokalen LUs ankommende Attach-Anforderungen akzeptiert, setzen Sie diesen Parameter auf einen leeren LU-Aliasnamen, indem Sie zwei Anführungszeichen (") angeben. Enthält die Datendatei für aufrufbare Transaktionsprogramme für einen TP-Namen mehr als einen Eintrag, kann nur für einen dieser Einträge ein leerer LU-Aliasname angegeben werden. Für alle anderen Einträge müssen verschiedene explizite LU-Aliasnamen ausgewiesen sein. Communications Server für Linux sucht für eine ankommende Attach-Anforderung für diesen TP-Namen ein TP mit einem passenden LU-Aliasnamen. Kann kein solches TP gefunden werden, wird nach einem TP mit einem leeren LU-Aliasnamen gesucht.

Wird in der Datei ein nicht aus Leerzeichen bestehender LU-Aliasname angegeben, muss das TP die erweiterte Form des APPC-Verbs RECEIVE\_ALLOCATE verwenden und diesen LU-Aliasnamen als Parameter für das Verb angeben. Dies ermöglicht Communications Server für Linux, die ankommende Attach-Anforderung an das richtige TP weiterzuleiten. Informationen zu den verschiedenen Formaten des Verbs RECEIVE\_ALLOCATE finden Sie im *Communications Server for Linux APPC Programmer's Guide*. Wenn das Transaktionsprogramm den richtigen LU-Aliasnamen nicht in die Anwendung integrieren, sondern zur Laufzeit bestimmen soll, können Sie eine Umgebungsvariable (mit dem Parameter ENV) auf den entsprechenden LU-Aliasnamen setzen und die Anwendung so konfigurieren, dass anhand dieser Umgebungsvariablen bestimmt wird, wie das Verb RECEIVE\_ALLOCATE abgesetzt werden soll.

Diese Zeile ist optional. Wenn sie nicht angegeben wird, werden standardmäßig ankommende Attach-Anforderungen von allen lokalen LUs akzeptiert, und das TP kann jede Form des APPC-Verbs RECEIVE\_ALLOCATE verwenden.

ENV Geben Sie alle für das Transaktionsprogramm erforderlichen Umgebungsvariablen an. Jede Variable wird in einer gesonderten, mit ENV beginnenden Zeile im Format *Umgebungsvariable=Wert* angegeben. Sie können bis zu 64 ENV-Zeilen aufnehmen. Die Variablen werden in der Reihenfolge gesetzt, in der sie hier aufgeführt sind.

Die Zeichenfolge *Umgebungsvariable=Wert* darf vor und nach dem Zeichen = keine Tabulatorzeichen oder Leerzeichen enthalten.

### WINDOWS

#### SHOW

Dieser Parameter wird nur angewendet, wenn die Anwendung eine grafische Benutzerschnittstelle hat. Handelt es sich um eine Konsolanwendung, wird der Parameter ignoriert. Geben Sie an, wie die Anwendung beim Start angezeigt werden soll. Dieser Parameter wird an die Anwendung übergeben und nicht von Communications Server für Linux verarbeitet. Die Anwendung ist für die Interpretation dieses Parameters und die entsprechende Reaktion auf den Parameter zuständig. Sie können einen der folgenden Werte angeben:

#### MAXIMIZED

Das Anwendungsfenster wird in maximaler Größe angezeigt.

#### MINIMIZED

Das Anwendungsfenster wird in Symbolgröße angezeigt.

**HIDDEN** Das Anwendungsfenster wird nicht auf dem Bildschirm angezeigt.

**NORMAL** Das Anwendungsfenster wird in normaler Größe an seiner normalen Position angezeigt.

#### NOACTIVATE

Das Anwendungsfenster wird in normaler Größe an seiner normalen Position angezeigt. Fokussiert bleibt jedoch das zuvor aktive Fenster. Das Fenster dieser Anwendung wird damit nicht zum aktiven Fenster.

#### MINNOACTIVATE

Das Anwendungsfenster wird in Symbolgröße angezeigt. Fokussiert bleibt jedoch das zuvor aktive Fenster.

Dieser Parameter ist optional. Wird er nicht aufgenommen, wird der Standardwert NORMAL verwendet.

#### SECURITY\_TYPE

Geben Sie wie folgt die Sicherheitseinstufung für die ausführbare Datei des TP an:

#### APPLICATION

Die ausführbare Datei des TP wird mit dem Systemaufruf `CreateProcess` als Anwendung gestartet.

#### SERVICE

Die ausführbare Datei des TP wird mit dem Systemaufruf `StartService` als Dienst gestartet. In diesem Fall muss der Dienst zuvor mit dem Dienstkontrollmanager unter dem vom Parameter `SERVICE_NAME` angegebenen Namen installiert worden sein.

Dieser Wert bezieht sich auf ein TP, das als ein Windows-Dienst ausgeführt wird (und nicht auf ein SNA-Service-TP, dessen Namen aus 4 Zeichen in Hexadezimalschreibweise besteht). Windows lässt nur die Ausführung einer Kopie eines Dienstes zu. Der Parameter `TYPE` sollte deshalb nicht auf `NON-QUEUED` gesetzt werden. Sollten Sie diesen Wert angeben, wird stattdessen der Wert `QUEUED-BROADCAST` verwendet.

### *SERVICE\_NAME*

Der Name des mit dem Dienstkontrollmanager installierten Dienstes. Dieser Parameter wird nur verwendet, wenn *SECURITY\_TYPE* auf *SERVICE* gesetzt ist.

### *USERID*

Geben Sie die Domäne und die Benutzer-ID an, die der Client zum Starten des TP verwenden soll, wenn *SECURITY\_TYPE* auf *APPLICATION* gesetzt ist. Dieser Parameter hat das Format *Domänename\Benutzer-ID*, wenn der Windows-Clientcomputer zu einer Domäne gehört, und *Computername\Benutzer-ID* (mit dem Namen des Windows-Clientcomputers an Stelle des Domänennamens), wenn der Windows-Clientcomputer nicht zu einer Domäne gehört.

Der Client versucht, das TP in der Anmeldesitzung des angegebenen Benutzers zu starten. Wenn das Feld *USERID* leer oder nicht angegeben ist, wird das TP in der Konsolsitzung gestartet. Falls der angegebene Benutzer nicht angemeldet oder kein Benutzer bei der Konsole angemeldet ist, wird das TP nicht gestartet. Der CS-Linux-Server wird über den gescheiterten Start informiert.



Beachten Sie die folgenden Hinweise zum Format dieser Einträge:

- Sie können eine Kommentarzeile aufnehmen. Geben Sie als erstes Zeichen dieser Zeile das Zeichen # ein. Communications Server für Linux wird diese Zeile ignorieren. Communications Server für Linux ignoriert außerdem Leerzeilen.
- Jeder Eintrag *Parameter=Wert* muss in einer Zeile stehen. Zeilenumbruchzeichen sind nicht zulässig. Eine Zeile darf maximal 255 Zeichen lang sein. Alle weiteren Zeichen werden ignoriert.
- Leerzeichen und Tabulatorzeichen am Anfang oder Ende der Zeile bzw. vor oder nach dem Zeichen = werden (außer bei der Zeichenfolge *Umgebungsvariable=Wert* für den Parameter *ENV*) ignoriert.
- Jede TP-Definition beginnt mit der Zeile, die den TP-Namen enthält, und endet mit der letzten Zeile der Definition oder bei der Zeile mit dem nächsten TP-Namen.
- Für ein TP sollten Sie jeden Parameter mit Ausnahme der Zeile *ENV*, die bis zu 64 Mal vorkommen kann, nur einmal angeben. Falls Sie einen Parameter mehrfach angeben, wird nur die letzte Instanz des jeweiligen Schlüsselworts verwendet.

## Format der Definitionsdatei für aufrufbares TP

---

## Anhang C. 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 IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Dienstleistungen von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Dienstleistungen können auch andere ihnen äquivalente Produkte, Programme oder Dienstleistungen verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb von Fremdprodukten, Fremdprogrammen und Fremddienstleistungen liegt beim Kunden.

Für in dieser Veröffentlichung beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieser Veröffentlichung 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 Cedex  
France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekanntgegeben. 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 dienen lediglich als Benutzerinformationen 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 unabhängigen, 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 Corporation  
P.O. Box 12195  
3039 Cornwallis Road  
Research Triangle Park, NC 27709-2195, USA

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 dieser Veröffentlichung aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der Allgemeinen Geschäftsbedingungen der IBM, der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer gesteuerten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen. Diese Daten stellen deshalb keine Leistungsgarantie dar.

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.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Diese Beispiele enthalten Namen von Personen, Firmen, Marken oder Produkten. 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. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, verwenden, vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Beispielprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. IBM kann deshalb nicht garantieren, dass die Zuverlässigkeit, Wartungsfreundlichkeit und Funktion dieser Programme gegeben ist. Sie dürfen diese Beispielprogramme kostenlos kopieren, Ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, verwenden, vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Beispielprogramme geschrieben werden.

Kopien oder Teile der Beispielprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten: Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corp. abgeleitet. ® Copyright International Business Machines Corporation. 1998, 2007. Alle Rechte vorbehalten.

---

## Marken

Folgende Namen sind Marken der IBM Corporation in den USA und/oder anderen Ländern:

Advanced Peer-to-Peer Networking	NetView
AIX	Operating System/2
Application System/400	Operating System/400
APPN	OS/2
AS/400	OS/400
CICS	PowerPC
DB2	PowerPC-Architektur
Enterprise System/3090	pSeries
Enterprise System/4381	S/390
Enterprise System/9000	System p5
ES/3090	System z
ES/9000	System/370
eServer	System/390
IBM	System 390
IBMLink	VSE/ESA
IMS	VTAM
MVS	WebSphere
MVS/ESA	z/OS
	z9

Folgende Namen sind Marken oder eingetragene Marken anderer Unternehmen:

Java und alle Java-basierten Marken sind Marken von Sun Microsystems, Inc. in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke und wird ausschließlich von The Open Group lizenziert.

Intel ist eine Marke der Intel Corporation.

Linux ist eine Marke von Linus Torvalds.

Microsoft, Windows, Windows 2003, Windows XP, Windows Vista und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

Andere Namen von Unternehmen, Produkten und Services können Marken oder Servicemarken anderer Unternehmen sein.



---

## Literaturverzeichnis

Die folgenden Veröffentlichungen von IBM enthalten zusätzliche Informationen zu den in diesem Handbuch behandelten Themen. Die Veröffentlichungen sind in folgende Themenbereiche unterteilt:

- Communications Server für Linux Version 6.2.3
- Systemnetzwerkarchitektur (SNA)
- Hostkonfiguration
- z/OS Communications Server
- Transmission Control Protocol/Internet Protocol (TCP/IP)
- X.25
- Advanced Program-to-Program Communication (APPC)
- Programmierung
- andere IBM Produkte für den Netzbetrieb

Zu den Handbüchern der CS-Linux-Bibliothek sind Kurzbeschreibungen vorhanden. Zu anderen Handbüchern sind nur Titel, Bestellnummer und in einigen Fällen der im vorliegenden Handbuch verwendete Kurztitel angegeben.

---

### Veröffentlichungen zu Communications Server für Linux Version 6.2.3

Die Bibliothek zu Communications Server für Linux enthält die nachstehend genannten Handbücher. Softcopies dieser Dokumentationen sind auf CD-ROM verfügbar. Informationen zum Aufrufen der Softcopy-Dateien von CD-ROM finden Sie in der Veröffentlichung *Einstieg in IBM Communications Server für Linux* (IBM Form GC12-3362). Für die Installation dieser Softcopy-Handbücher benötigen Sie (in Abhängigkeit von den installierten Landessprachen) 9-15 MB Festplattenspeicherplatz.

- *Einstieg in IBM Communications Server für Linux* (IBM Form GC12-3362)  
Dieses Handbuch enthält eine allgemeine Einführung in Communications Server für Linux und Informationen zu unterstützten Netzkenndaten sowie zu Installation, Konfiguration und Betrieb.
- *IBM Communications Server für Linux Verwaltungshandbuch* (IBM Form SC12-3363)  
Dieses Handbuch gibt einen Überblick über SNA und Communications Server für Linux und enthält Informationen zu Konfiguration und Betrieb von Communications Server für Linux.
- *IBM Communications Server for Linux Administration Command Reference* (IBM Form SC31-8587)  
Dieses Handbuch enthält Informationen zu SNA- und CS-Linux-Befehlen.
- *IBM Communications Server for Linux CPI-C Programmer's Guide* (IBM Form SC31-6774)  
Dieses Handbuch enthält Informationen für erfahrene "C"- oder Java-Programmierer zum Schreiben von SNA-Transaktionsprogrammen mit der CS-Linux-API CPI Communications.
- *IBM Communications Server for Linux APPC Programmer's Guide* (IBM Form SC31-6773)

Dieses Handbuch enthält die für das Schreiben von Anwendungsprogrammen mit APPC (Advanced Program-to-Program Communication) erforderlichen Informationen.

- *IBM Communications Server for Linux LUA Programmer's Guide* (IBM Form SC31-6776)

Dieses Handbuch enthält die für das Schreiben von Anwendungen mit der herkömmlichen Anwendungsprogrammierschnittstelle für LU-Anwendungen LUA erforderlichen Informationen.

- *IBM Communications Server for Linux CSV Programmer's Guide* (IBM Form SC31-6775)

Dieses Handbuch enthält die für das Schreiben von Anwendungsprogrammen mit der API (Anwendungsprogrammierschnittstelle) Common Service Verbs (CSV) erforderlichen Informationen.

- *IBM Communications Server for Linux MS Programmer's Guide* (IBM Form SC31-6777)

Dieses Handbuch enthält die für das Schreiben von Anwendungen mit der API Management Services (MS) erforderlichen Informationen.

- *IBM Communications Server for Linux NOF Programmer's Guide* (IBM Form SC31-8595)

Dieses Handbuch enthält die für das Schreiben von Anwendungen mit der API Node Operator Facility (NOF) erforderlichen Informationen.

- *IBM Communications Server for Linux Diagnostics Guide* (IBM Form SC31-8588)

Dieses Handbuch enthält Informationen zur Behebung von SNA-Netzfehlern.

- *IBM Communications Server for Linux APPC Application Suite User's Guide* (IBM Form SC31-6772)

Dieses Handbuch enthält Informationen zu APPC-Anwendungen, die zusammen mit Communications Server für Linux eingesetzt werden können.

- *IBM Communications Server für Linux Glossar* (IBM Form GC12-3365)

Dieses Handbuch enthält eine umfangreiche Liste von Begriffen und Definitionen, die in allen Veröffentlichungen der Bibliothek zu IBM Communications Server für Linux verwendet werden.

---

## Veröffentlichungen zur Systemnetzwerkarchitektur (SNA)

Die folgenden Handbücher enthalten Informationen zu SNA-Netzen:

- *Systems Network Architecture: Format and Protocol Reference Manual—Architecture Logic for LU Type 6.2* (IBM Form SC30-3269)
- *Systems Network Architecture: Formats* (IBM Form GA27-3136)
- *Systems Network Architecture: Guide to SNA Publications* (IBM Form GC30-3438)
- *Systems Network Architecture: Network Product Formats* (IBM Form LY43-0081)
- *Systems Network Architecture: Technical Overview* (IBM Form GC30-3073)
- *Systems Network Architecture: APPN Architecture Reference* (IBM Form SC30-3422)
- *Systems Network Architecture: Sessions between Logical Units* (IBM Form GC20-1868)
- *Systems Network Architecture: LU 6.2 Reference—Peer Protocols* (IBM Form SC31-6808)
- *Systems Network Architecture: Transaction Programmer's Reference Manual for LU Type 6.2* (IBM Form GC30-3084)
- *Systems Network Architecture: 3270 Datastream Programmer's Reference* (IBM Form GA23-0059)

- *Networking Blueprint Executive Overview* (IBM Form GC31-7057)
- *Systems Network Architecture: Management Services Reference* (IBM Form SC30-3346)

---

## Veröffentlichungen zur Hostkonfiguration

Die folgenden Handbücher enthalten Informationen zur Hostkonfiguration:

- *ES/9000, ES/3090 IOCP User's Guide Volume A04* (IBM Form GC38-0097)
- *3174 Establishment Controller Installation Guide* (IBM Form GG24-3061)
- *3270 Information Display System 3174 Establishment Controller: Planning Guide* (IBM Form GA27-3918)
- *OS/390 Hardware Configuration Definition (HCD) User's Guide* (IBM Form SC28-1848)

---

## Veröffentlichungen zu z/OS Communications Server

Die folgenden Handbücher enthalten Informationen zu z/OS Communications Server:

- *z/OS V1R7 Communications Server: SNA Network Implementation Guide* (IBM Form SC31-8777)
- *z/OS V1R7 Communications Server: SNA Diagnostics* (Band 1: IBM Form GC31-6850, Band 2: IBM Form GC31-6851)
- *z/OS V1R6 Communications Server: Resource Definition Reference* (IBM Form SC31-8778)

---

## Veröffentlichungen zu TCP/IP

Die folgenden Handbücher enthalten Informationen zum Netzprotokoll TCP/IP (Transmission Control Protocol/Internet Protocol):

- *z/OS V1R7 Communications Server: IP Configuration Guide* (IBM Form SC31-8775)
- *z/OS V1R7 Communications Server: IP Configuration Reference* (IBM Form SC31-8776)
- *z/VM V5R1 TCP/IP Planning and Customization* (IBM Form SC24-6125)

---

## Veröffentlichungen zu X.25

Die folgenden Handbücher enthalten Informationen zum Netzprotokoll X.25:

- *Communications Server for OS/2 Version 4 X.25 Programming* (IBM Form SC31-8150)

---

## Veröffentlichungen zu APPC

Die folgenden Handbücher enthalten Informationen zu APPC (Advanced Program-to-Program Communication):

- *APPC Application Suite V1 User's Guide* (IBM Form SC31-6532)
- *APPC Application Suite V1 Administration* (IBM Form SC31-6533)
- *APPC Application Suite V1 Programming* (IBM Form SC31-6534)
- *APPC Application Suite V1 Online Product Library* (IBM Form SK2T-2680)
- *APPC Application Suite Licensed Program Specifications* (IBM Form GC31-6535)
- *z/OS V1R2.0 Communications Server: APPC Application Suite User's Guide* (IBM Form SC31-8809)

---

## Veröffentlichungen zur Programmierung

Die folgenden Handbücher enthalten Informationen zur Programmierung:

- *Common Programming Interface Communications CPI-C Reference* (IBM Form SC26-4399)
- *Communications Server for OS/2 Version 4 Application Programming Guide* (IBM Form SC31-8152)

---

## Veröffentlichungen zu anderen IBM Produkten für den Netzbetrieb

Die folgenden Handbücher enthalten Informationen zu anderen Themen mit Bezug zu Communications Server für Linux:

- *SDLC Concepts* (IBM Form GA27-3093)
- *Local Area Network Concepts and Products: LAN Architecture* (IBM Form SG24-4753)
- *Local Area Network Concepts and Products: LAN Adapters, Hubs and ATM* (IBM Form SG24-4754)
- *Local Area Network Concepts and Products: Routers and Gateways* (IBM Form SG24-4755)
- *Local Area Network Concepts and Products: LAN Operating Systems and Management* (IBM Form SG24-4756)
- *IBM Network Control Program Resource Definition Guide* (IBM Form SC30-3349)

---

# Index

## Numerische Stichwörter

- 3270
  - Arbeitsblatt 186
  - LU-Konfiguration 81
  - Pool-Konfiguration 83
- 5250
  - Arbeitsblatt 185

## A

- Abhängige LU
  - Beschreibung 10
  - konfigurieren 81
- Abhängiger Knoten 3
- Adapterkartenummer (Parameter) 65
- Adresse des Zielhosts (Parameter) 121
- Adressierbare Netzeinheit (NAU) 6
- Advanced Peer-to-Peer Networking (APPN) 1
- Aktiv bei Initialisierung (Parameter) 65, 79
- Aktivieren der SNA-Software
  - Remote API Client unter AIX oder Linux 159
- Aktivieren der Software Communications Server für Linux
  - auf einem Server 40
  - Fehler bei der Initialisierung 42
  - Remote API Client unter Windows 144
- Aktivierung (Parameter) 71
- Aliasname (Parameter) 91
- Aliasname, Partner-LU 90
- Aliasname der lokalen LU (Parameter) 106
- Aliasname für Steuerpunkt (Parameter) 59
- Allgemeiner Datenstrom (GDS) 7
- Anfangswert (Parameter) 104
- Anforderungseinheit (RU) 104
- Ankommende Allocate-Anforderungen an aktives TP weiterleiten (Parameter) 97
- Ankommende Zuordnungen in Warteschlange stellen (Parameter) 97
- ANR
  - Beschreibung 13, 21
  - dynamische Weiterschaltung 25
- Anwendung
  - Arbeitsblätter 165
  - Programm 6
- Anwendungsprogrammierschnittstelle (API) 6
- Anwendungsunterstützung, Arbeitsblätter 181
- Anzeige-LU (Parameter) 118
- API
  - Beschreibung 6
  - privat 6

- API (*Forts.*)
  - zu Communications Server für Linux 6
- API-Trace
  - Remote API Client unter Windows 153
- APPC
  - Arbeitsblatt 181
  - Konfiguration 85
  - Sicherheit 108
- APPCLLU
  - Remote API Client unter Windows 158
- APPCTPN
  - Remote API Client unter Windows 158
- Application System/400 (AS/400) 14
- APPN
  - Beispielnetz 14
  - Beschreibung 1, 14
  - Endknoten 5, 16, 166
  - Knotentypen 14
  - Netz 13, 14, 25
  - Netzknoten 5, 15, 165
  - Routenauswahl 24
  - Steuerpunkt 16
  - Verbindungsnetz 27
  - Zweignetzknoten 5, 167
- APPN-Unterstützung (Parameter) 59
- Arbeitsblätter 39
- Argumente (Parameter) 98
- Art der Zweigverbindung (Parameter) 75
- Art des fernen Knotens (Parameter) 74
- AS/400 (Application System/400) 14
- Auf SNA-definierte Werte zurücksetzen (Parameter) 105
- Auf Verbindungsnetz definieren (Parameter) 67, 68
- Aufgabenblätter 39
- Aufrufadresse (Parameter) 72
- Aufrufbares Transaktionsprogramm 11
  - Datendatei 35
  - für Communications Server für Linux definieren 94
  - snatpinstall verwenden 189
- Aufrufendes Transaktionsprogramm 11, 94
- Austausch formatierter Daten 12
- Ausweichserver 137
- Ausweichserver für Hauptserver 57
- Automatic Network Routing (ANR) 13, 21
- Automatisch aktivierte Sitzungen (Parameter) 103

## B

- Befehle
  - Konfigurationsserver ändern 58

- Befehlszeilenverwaltungsprogramm
  - Befehlstypen 55
  - Beschreibung 34
  - Hilfe 55
  - verwenden 55
  - von einem Client 55
- Benutzer-ID (Parameter) 98, 107, 109
- Benutzer in Zugriffsliste (Parameter) 110
- BIND-Anforderung 10
- Branch Extender 28

## C

- CICS (Customer Information Control System) 8
- Client
  - aufrufbares Transaktionsprogramm
    - konfigurieren 189
    - Netzdatendatei 36
    - Parameter ARGUMENTS 192
    - Parameter PATH 192
    - Parameter SECURITY\_TYPE 196
    - Parameter SERVICE\_NAME 197
    - Parameter SHOW 196
    - Parameter TIMEOUT 193
    - Parameter TP-Name 191
    - Parameter TYPE 192
    - TP definieren 164
    - verwalten 137
    - Voraussetzung für den Netzbetrieb 139
  - Client/Server
    - Konfiguration 57
    - Trace 155, 160
- Clustercontroller 3
- COS
  - Beschreibung 13
  - Typen 101
  - Zweck 100
- CP (Control Point) 8, 9
- CP-CP-Sitzung 10
- CPI-C (Common Programming Interface for Communications)
  - Arbeitsblatt 185
  - Nebeninformationen 105
- CPI-C-Nebeninformationen
  - Konfigurationsmethoden 105
  - Parameter 106, 107
  - zusätzlicher Konfigurationsbedarf 108
- CSVTBLG
  - Remote API Client unter Windows 158
- Customer Information Control System (CICS) 8

## D

- Datendatei
    - aufrufbares Transaktionsprogramm 35
    - Clientnetz 36
    - Domänenkonfiguration 35
    - Knotenkonfiguration 35
    - SNA-Netz 36
    - TP-Definition 35
  - Datenübertragungssteuerung (DLC) 63
  - DCA (Document Content Architecture) 8
  - Dependent LU Server (DLUS) 30
  - DFV-Controller 3
  - DFV-Verbindung 3
  - Diagnosetools 37
  - Dialog
    - Beschreibung 11
    - Sicherheit 109
  - Dialoganzeige 44
  - Dialogart (Parameter) 100
  - Dialogsicherheit
    - Konfigurationsmethoden 109
    - Parameter 109
  - Direkte Konnektivität 25
  - DLC
    - Konfiguration 63, 65
    - Konfigurationsmethoden 65
    - zusätzlicher Konfigurationsbedarf 69
  - DLUR
    - Arbeitsblatt 178
    - Beschreibung 30
    - Konfiguration 124
    - zusätzlicher Konfigurationsbedarf 80
  - DLUR für untergeordnete Knoten 79
  - DLUR-PU
    - Konfigurationsmethoden 78
    - Parameter 78, 79
  - DLUS
    - Beschreibung 30
  - DLUS-Name (Parameter) 78, 79
  - Document Content Architecture (DCA) 8
  - Domäne
    - Beschreibung 3
    - Konfigurationsdatei 35
  - Domänenfenster 45
  - Domänenname
    - ändern 138
  - Domänenressourcen 57
  - Drucker 4
  - Drucker-LU (Parameter) 118
  - Durchgriff-DLUR 79
  - Durchgriffsdienste
    - Arbeitsblätter 165, 177
    - konfigurieren 113
- ## E
- Einleitende Sitzungsbegrenzung (Parameter) 102
  - Endknoten 5
    - APPN 16
    - Beschreibung 5
    - im APPN-Beispielnetz 15
    - Verzeichnis 18, 19

- Enterprise Extender (HPR/IP)
  - Arbeitsblatt 176
  - Port-Konfiguration 64
- ENV (Parameter) 195
- Escape-Zeichen, RCF 129
- Ethernet
  - Arbeitsblatt 172
  - Port-Konfiguration 64
- Ethernet-Typ (Parameter) 68

## F

- Fehlerbehebung 37
- Fehlerbehebung auf Verbindungsebene für implizite Verbindungen (Parameter) 69
- Fehlerprotokolldatei 37
- Fenster
  - Beschreibung 44
  - CPI-C-Bestimmungsorte 45
  - Domäne 44, 45
  - Funktionsleistungsflächen 49
  - Knoten 44, 46
  - LU-Pools 45
  - Menüs 44
  - Ressource 44
  - Ressourceneinträge 49
- Fenster für Empfangsnachrichtendosierung (Parameter) 103
- FEP (Front-End Processor) 3
- Ferne X.25-Adresse (Parameter) 73
- Ferner Knoten
  - definieren 88
  - Konfigurationsmethoden 89
  - LU 9
  - Parameter 'SNA-Netzname des Knotens' 89
  - Partner-LU 90
  - zusätzlicher Konfigurationsbedarf 89
- Formate 1
- Formatfreier Datenaustausch 12
- Front-End-Prozessor (FEP) 3
- Funktion des fernen Knotens (Parameter) 76
- Funktion für ferne Befehle (RCF) 35

## G

- GDS (General Data Stream) 7
- Gemischte Netze 2, 30
- Generischer Name der Partner-LU (Parameter) 90
- Gezielte Suche 21
- Grenzknoten 3
- GROUP (Parameter) 194
- Gruppen-ID (Parameter) 98

## H

- Hauptserver 57, 137
  - angeben 138
- High-Performance Routing (HPR) 13, 21
- Hilfe
  - Befehlszeilenverwaltungsprogramm 55
  - Motiv-Verwaltungsprogramm 54

- Host 3
- Hostknoten 3
- Hostverbindungsstation/DLUR-PU (Parameter) 82, 87
- HPR
  - Beschreibung 13, 21
  - HPR auf impliziten Verbindungen unterstützt (Parameter) 69
- HTTPS 142

## I

- ID des fernen Knotens (Parameter) 75
- ID des lokalen Knotens (Parameter) 75
- IMS/VS (Information Management System/Virtual Storage) 8
- Inaktivieren der Software 42
- Inaktivieren der Software Communications Server für Linux
  - Remote API Client unter Windows 144
- Information Management System/Virtual Storage (IMS/VS) 8
- Interner Trace
  - Remote API Client unter Windows 157
- IP-Adressformate 139
- IP-Port-Nummern 140
- IPV4-Adresse 139
- IPV6-Adresse 139
- ISR 21, 25

## J

- Jobferneingabe (RJE) 8

## K

- Kennwort (Parameter) 107, 109
- Kernel-Komponenten, Trace durchführen 41
- Kernel-Speicherbegrenzung 41
- Knoten
  - Arbeitsblätter 165
  - Konfigurationsdatei 35
  - Konfigurationsmethoden 58
  - Parameter 59
  - Peer 2
  - peripher 2
  - SNA 2
  - Typen 2, 5
  - Unterbereichs-SNA 2
  - zusätzlicher Konfigurationsbedarf 59
  - Zweck 58
- Knoten-ID (Parameter) 59
- Knoten mit DFV-Controller 3
- Knotenfenster 46
- Knotenressourcen 57
- Knotentyp 2.0 3
- Knotentyp 2.1 3
- Knotentyp 4 3
- Knotentyp 5 2
- Komprimierung unterstützt (Parameter) 79, 105
- Konfiguration
  - abhängige LU 81

- Konfiguration (*Forts.*)
  - APPC-Kommunikation 85
  - APPC-Sicherheit 108
  - Aufgaben 57
  - CPI-C-Nebeninformationen 105
  - Dateien 35
  - DLC 64
  - DLUR 124
  - Durchgriffsdienste 113
  - Knoten 58
  - Konnektivität 63
  - Port 64
  - Sicherheitszugriffsliste 110
  - SNA Gateway 122
  - TN-Redirector-Zugriffseinträge 118
  - TN-Server-Zugriffseinträge 114
  - TN-Server-Zuordnungseinträge 117
  - TP 93
  - Verbindungsnetz 64
- Konfigurationsserver 57
  - entfernen 58
  - hinzufügen 58
- Konnektivität
  - Arbeitsblätter 165, 168
  - Beschreibung 5
  - direkt 25
  - Konfiguration 63
- Kurze Protokollierung 37

## L

- LAN-Trace
  - auf einem Client 160
- LAN-Zugriff, Zeitlimit 141
- Leitungsangaben (Parameter) 66
- LEN-Knoten 5
  - Arbeitsblatt 167
  - Beschreibung 5, 16
  - Features 14
  - Verzeichnis 18, 19
- Linux-Befehle 127
- Linux-Client
  - Domänenname 161
  - maximum\_element\_count 161
  - maximum\_header\_count 161
  - maximum\_process\_count 161
- Logische Einheit (LU) 7
- Logischer Satz 12
- Lokale IP-Schnittstelle (Parameter) 68
- Lokale LU
  - Beschreibung 9
  - definieren 86
  - Konfigurationsmethoden 86
  - Parameter 87
  - zusätzlicher Konfigurationsbedarf 87
- Lokale LU (Parameter) 106, 108
- Lokale SAP-Nummer (Parameter) 67
- Lokale Topologiedatenbank 22
- Lokaler Knoten
  - LU 9
- Lokalisieren von Ressourcen 17
- LU
  - Beschreibung 7
  - Typen 7
- LU 0
  - Beschreibung 8
- LU 1 8

- LU 2 8
- LU 3 7
- LU 6.2
  - Beschreibung 7
  - Konfiguration 85
- LU-Aliasname (Parameter) 87, 96
- LU-Datenverkehr (Parameter) 72
- LU im Pool (Parameter) 83
- LU-LU-Sitzung 9
- LU-Name (Parameter) 82, 87
- LU-Nummer (Parameter) 82, 87, 124
- LU-Pool
  - anzeigen 84
  - definieren 83
  - Konfigurationsmethoden 84
  - Parameter 84
- LU-Typ (Parameter) 82
- LU-Typen 0-3
  - Konfigurationsmethoden 81
  - Parameter 82, 83
  - zusätzlicher Konfigurationsbedarf 83
- LUA
  - Arbeitsblatt 187
  - Konfiguration 81
  - Pool-Konfiguration 83
- LUALIAS (Parameter) 195

## M

- MAC (Medium Access Control) 28
- MAC-Adresse (Parameter) 73
- Management Services (API) 6
- Management Services (MS) 15
- Max. Anzahl aktiver Schabloneninstanzen (Parameter) 69
- Max. RU-Größe einschränken (Parameter) 104
- Maximale RU-Größe (Parameter) 104
- Maximale Sitzungsbegrenzung (Parameter) 103
- Maximum (Parameter) 104
- Medium Access Control (MAC) 28
- Mehrfachsitzungen 11
- Mindestanzahl Sitzungen für Konfliktgewinner (Parameter) 103
- Mindestanzahl Sitzungen für Konfliktverlierer (Parameter) 103
- Mit den Parametern kann jede LU aufgerufen werden (Parameter) 96
- Modus 100
  - Beschreibung 12
  - Konfiguration 102
  - Parameter 102, 103, 104, 105
  - Standard 101
  - zusätzlicher Konfigurationsbedarf 105
- Modus (Parameter) 106
- Motif-Verwaltungsprogramm
  - aufrufen 43
  - Beschreibung 33
  - Dialoganzeige 51, 53
  - Domänenfenster 45
  - Funktionsleistungsflächen 49
  - Hilfe 54
  - Knotenfenster 46
  - Ressourceneinträge 49
  - Ressourcenfenster 44

- Motif-Verwaltungsprogramm (*Forts.*)
  - verwenden 43
- MPC
  - Port-Konfiguration 64
- MPC-Gruppe (Parameter) 73
- MS (Management Services) 15
- Multipath Channel
  - Arbeitsblatt 175
- MultiPath Channel (MPC)
  - Port-Konfiguration 64

## N

- Nachbarknoten 14
- Name (Parameter)
  - LU-Pool 84
  - Modus 102
  - Sicherheitszugriffsliste 110
  - symbolischer CPI-C-Bestimmungs-ort 106
  - Verbindungsstation 71
- Name der lokalen LU (Parameter) 92
- Name der Partner-LU (Parameter) 90, 92
- Name der übergeordneten LU (Parameter) 124
- Name der untergeordneten LU (Parameter) 124
- Name der untergeordneten PU (Parameter) 76, 79
- Name der Verbindungsstation (Parameter) 92
- Name des Ausweich-DLUS (Parameter) 78
- Name des fernen Knotens (Parameter) 74
- Name des Transaktionsprogramms (Parameter) 96, 99
- Name des übergeordneten DLUS (Parameter) 76
- Name des Verbindungsnetzes (Parameter) 68
- NAP (Network Access Process) 143
- NAU (Network Accessible Unit) 6
- NetView
  - Ändern der Größe des Eingabebereichs 128
  - Befehle 127
  - Beschreibung 127
  - Bildschirmanzeige 128
  - Programm 127
  - Servicepunkt 127
  - Versionsnummern 127
- Netz
  - gemischt 30
  - Topologiedatenbank 21
  - Typen 2
  - Verwaltung 127
- Netzdatendatei
  - Beschreibung 36
  - Remote API Client unter AIX oder Linux 160
- Netzknoten 5
  - Beispielkonfiguration 15
  - Verzeichnis 18, 20
- Netzknotenserver 5, 15
- Netzverwaltungsdaten 10
- Netzzugriffsprozess (NAP) 143

Nicht interpretierter Name (Parameter) 91  
Node Operator Facility (API) 6  
NOF (Node Operator Facility), API 36

## P

Parallelsitzungen 11  
Partner-LU 9  
    Aliasnamen definieren 90  
    auf fernem Knoten definieren 90  
    Konfigurationsmethoden 90  
    mehrere mit Platzhaltern definieren 90  
    Parameter 90, 91  
    zusätzlicher Konfigurationsbedarf 93  
Partner-LU (Parameter) 106, 108  
Partner-TP (Parameter) 106  
Peer-zu-Peer-Kommunikation 1  
Peernetz 2  
    Knotentypen 5  
    Routenauswahl 13  
Peripherer Knoten 3  
Pfad für ausführbare CS-Linux-Programme 40  
Physische Einheit (PU) 7  
PIP erlaubt (Parameter) 100  
Planungsblätter 39  
Platzhalter 90  
Poolname (Parameter) 83  
Port  
    Konfiguration 64, 65  
    Parameter 65, 66, 67, 68, 69  
    zusätzlicher Konfigurationsbedarf 69  
Port-Nummer (Parameter) 65  
Position (Parameter) 91  
Primäre LU 10  
Protokoll (Parameter) 68  
Protokolldateien  
    konfigurieren 60  
    Typen 60  
Protokolle 1  
Protokollierung 61  
    Remote API Client unter Windows 150  
Protokollnachrichten 37  
Prüfprotokolldatei 37  
PU  
    Beschreibung 7  
    für DLUR 77  
PU-ID (Parameter) 78  
PU-Name (Parameter) 78  
PUCP (Physical Unit Control Point) 9

## Q

QLLC  
    Arbeitsblatt 174  
    Port-Konfiguration 64  
Quellen-TP 11, 94

## R

Rapid Transport Protocol (RTP) 13, 21  
RCF  
    Befehlssyntax 128

RCF (*Forts.*)  
    Funktionen 35  
    gültige Zeichen 129  
Remote API Client für AIX oder Linux  
    Servernamen 163  
Remote API Client unter AIX oder Linux  
    \* 162  
    broadcast\_attempt\_count 162  
    invoked\_tps 161  
    lan\_access\_timeout 161  
    server\_lost\_timeout 162  
    Verwaltung 159  
Remote API Client unter Windows  
    admin\_msg 156  
    aktivieren 144  
    all\_api 154  
    appc 154  
    APPCLU 158  
    APPCTPN 158  
    audit\_file 152  
    audit\_file\_wrap\_size 152  
    audit\_logging\_enabled 151  
    aufgerufene TPs 147  
    backup\_audit\_file 152  
    backup\_error\_file 151  
    broadcast\_attempt\_count 148  
    client\_start\_timeout 148  
    CPI-C-Anwendungsdaten 158  
    cpic 155  
    csv 155  
    CSV-Anwendungsdaten 158  
    CSVTBLG 159  
    data 156  
    datagram 156  
    Domäne 147  
    error\_file 151  
    error\_file\_wrap\_size 152  
    exception\_logging\_enabled 151  
    file1 153  
    file1 (CS\_tracing) 155  
    file1 (Internal\_tracing) 157  
    file2 154  
    file2 (CS\_tracing) 155  
    file2 (Internal\_tracing) 157  
    flip\_size 154  
    flip\_size (CS\_tracing) 156  
    flip\_size (Internal\_tracing) 157  
    inaktivieren 144  
    Informationen zum API-Trace 153  
    Informationen zum Client-Server-Trace 155  
    Informationen zum internen Trace 157  
    Informationen zur Protokollierung 150  
    Konfiguration 145  
    Konfigurationsdaten 147  
    lan\_access\_timeout 148  
    log\_directory 151  
    maximum\_element\_count 147  
    maximum\_header\_count 147  
    maximum\_process\_count 147  
    nof 155  
    receive 156  
    rui 155  
    send 156  
    server\_lost\_timeout 148

Remote API Client unter Windows (*Forts.*)  
    Server1 149  
    Server2 - Server9 150  
    Serverinformationen 149  
    Status 144  
    succinct\_audits 153  
    succinct\_errors 153  
    trace\_flushing 157  
    trace\_level 157  
    truncation\_length 154  
Ressourcen lokalisieren 17  
Ressourcennamen 17  
RJE (Remote Job Entry) 8  
Route 13  
Routenauswahl 13, 21, 24  
Routenwahl für Verbindungsstationen  
    Parameter 92  
RTP  
    Beschreibung 13, 21  
    Endpunkte 25  
RU (Request Unit) 104

## S

SAP (Service Access Point) 28  
SAP-Nummer (Parameter) 73  
SATF  
    direkte Konnektivität 25  
    im APPN-Netz 27  
SDLC  
    Arbeitsblatt 168  
    Port-Konfiguration 64  
Secure Sockets Layer (SSL) 116, 120  
    Clientauthentifizierung 116, 120  
    Datenverschlüsselung 116, 120  
    Serverauthentifizierung 117, 120, 122  
Sekundäre LU 10  
SEND (Funktion) 12  
Server  
    aktivieren 40  
    Beziehung zum Client 137  
    entfernen 58  
    hinzufügen 58  
    inaktivieren 42  
Serviceklasse (COS) 13  
Serviceklassenname (Parameter) 102  
Servicepunkt 127  
Servicepunkt-Befehlsfunktion (SPCF) 35, 127  
Servicezugriffspunkt (SAP) 28  
Shared-Access Transport Facility (SATF) 25  
Sicherheit  
    APPC 108  
    Dialog 109  
    Sitzung 108  
    UCF 132, 134  
Sicherheit (Parameter) 107  
Sicherheit auf Dialogebene erforderlich (Parameter) 100  
Sicherheitszugriffsliste  
    Konfigurationsmethoden 110  
    Parameter 110  
    zusätzlicher Konfigurationsbedarf 110  
Zweck 110

Sicherheitszugriffsliste (Parameter) 100  
 Sitzung  
   Beschreibung 9  
   Routing 21  
   Typen 9  
 Sitzungssicherheit  
   Konfigurationsmethoden 108  
   Parameter 108, 109  
   zusätzlicher Konfigurationsbedarf 109  
 Sitzungszeitlimit (Parameter) 104  
 SNA  
   APPN-Konzepte 14  
   Basiskonzepte 2  
   Beschreibung 1  
   hierarchische Struktur 2  
   Netz 1  
   Netzdatendatei 36, 160  
   Netztypen 2  
   Schichten 2  
   Unterbereichs-SNA 1  
 sna\_clnt.net (Datei) 160  
 SNA Gateway  
   Arbeitsblatt 179  
   Zweck 122  
 SNA-Netzdaten  
   Remote API Client unter Windows 145  
 SNA-Netzname des Knotens (Parameter) 89  
 SNA-Port-Name (Parameter) 65, 71  
 snaadmin (Programm) 34  
 snanetutil (Programm) 138  
 SPCF  
   Befehle 130  
   Befehlssyntax 128  
   Beschreibung 35, 127  
 SSCP (System Services Control Point) 8  
 SSCP-abhängige LU 10  
 SSCP-LU-Sitzung 9  
 SSCP-PU-Sitzung 10  
 Standard-LU verwenden (Parameter) 106  
 Standardausgabe (Parameter) 99  
 Standardeingabe (Parameter) 99  
 Standardfehler (Parameter) 99  
 start (Befehl) 41  
 Status  
   Remote API Client unter Windows 144  
 Steuerdaten 10  
 Steuerpunkt (CP) 8, 9  
 Steuerpunkt der physischen Einheit (PUCP) 9  
 Steuerpunkt für Systemdienste (SSCP) 8  
 Steuerpunktname (Parameter) 59  
 stop (Befehl) 43  
 Suche über Broadcast-Betrieb 21  
 Synchronisationsebene (Parameter) 100

## T

TCP/IP-Port-Nummer (Parameter) 115, 119, 121  
 Teil des Standardpools (Parameter) 87  
 Telnet-Clientadresse (Parameter) 119  
 Terminal 3

Terminalcontroller 3  
 TN Redirector  
   Arbeitsblatt 180  
   Parameter für Zugriffseintrag 119, 121  
   Zugriffseintrag 118  
 TN Server  
   Arbeitsblatt 179  
   Parameter für Zugriffseintrag 115, 116  
   Parameter für Zuordnungseintrag 118  
   Zugriffseintrag 114, 117  
   Zuordnungseintrag 117  
 TN3270-Clientadresse (Parameter) 115  
 TN3270E unterstützen (Parameter) 115  
 Token-Ring  
   Arbeitsblatt 170  
   Port-Konfiguration 64  
 Topologie- und Routing-Dienste (TRS) 21  
 Topologiedaten 10  
   lokal 22  
   Verbindungsnetz 22  
 Topologiedatenbank aktualisieren 23  
 TP  
   APPC-Definitionsparameter 99, 100  
   aufrufbar 11, 94  
   aufrufen 11, 94  
   Aufrufparameter 96, 97, 98, 99  
   Beschreibung 5  
   Client 164  
   Konfiguration 93  
   Konfigurationsmethoden 95  
   Quelle 11, 94  
   Ziel 11  
 Trace  
   Client/Server 160  
   LAN 160  
 Trace-Datei 37  
 Trace für Kernel-Komponenten durchführen 41  
 Transaktionsprogramm 5  
   Ziel 94  
 Transaktionsprogramm, Konfigurationsparameter  
   ENV 195  
   GROUP 194  
   LUALIAS 195  
   USERID, AIX oder Linux 194  
   USERID, Windows 197  
 Transportnetz 13  
 TRS (Topology and Routing Services) 21

## U

UCF  
   Abbrechen eines Befehls 134  
   Ausgabe 133  
   Befehlssyntax 128, 132  
   Beispielbefehl 133  
   Benutzer 131  
   Benutzername 134  
   Berechtigungen 132  
   Beschreibung 35, 127  
   Dämonprogramm 131  
   gültige Befehle 132

UCF (Forts.)  
   Sicherheit 132, 134  
   verwenden 131  
   Zugriff auf Dateien 134  
   zulässige Befehle 132  
 UDP/IP-Übertragung 140  
 Umgebung (Parameter) 99  
 Unabhängige LU  
   Beschreibung 10  
   Konfiguration 85  
 Unbegrenzte Verbindungsversuche zum DLUS (Parameter) 79  
 UNIX-Befehlsfunktion (UCF) 35  
 Unterbereichs-SNA 1  
 Unterbereichsknoten 2  
 Unterbereichsnetz  
   Beispiel 4  
   Beschreibung 2  
   Knotentypen 2  
   Routenauswahl 13  
 Untergeordnete LUs für SNA Gateway  
   Konfigurationsmethoden 123  
   Parameter 124  
   zusätzlicher Konfigurationsbedarf 124  
 Untergeordnete LUs für Zugriff auf implizite PUs konfigurieren (Parameter) 69  
 Unterstützt Parallelsitzungen (Parameter) 91  
 Unterstützung für Fehlerbestimmung  
   Protokollierung 60  
   Überblick 37  
 USERID (Parameter)  
   AIX oder Linux 194  
   Windows 197  
 ux-cancel (Befehl) 134

## V

Verbindungsgruppe 21  
 Verbindungsnetz 14  
   APPN 27  
   Beschreibung 14  
   Konfiguration 64, 65  
   Konfigurationsmethoden 65  
   Topologiedaten 22  
   zusätzlicher Konfigurationsbedarf 69  
 Verbindungsstation 69  
   Beschreibung 5  
   Konfiguration 69, 70  
   Parameter 71, 72, 73, 74, 75, 76  
   zusätzlicher Konfigurationsbedarf 77  
 Version der IP-Adresse 139  
 Versionsnummern von NetView 127  
 Verwaltung  
   Tools 32  
   Zuständigkeit 31  
 Verzeichnis  
   Endknoten 18, 19  
   für ausführbare CS-Linux-Programme 40  
   LEN-Knoten 18, 19  
   Netzknoten 18, 20  
 Verzögerte Anmeldung (Parameter) 124  
 Virtual Terminal Access Method (VTAM) 14

Virtueller Routing-Knoten (VRN) 27  
Vollständig qualifizierter CP-Name 17  
Vollständig qualifizierter LU-Name 18  
Vollständige Protokollierung 37  
Vollständiger Pfad zur ausführbaren  
  Datei des Transaktionsprogramms  
  (Parameter) 98  
VRN  
  Beschreibung 27  
VTAM (Virtual Terminal Access  
  Method) 14

## W

WebSphere Application Server 142  
Weitervermittlung einer Sitzungsroute  
  (ISR) 21, 25  
Windows-Client  
  Netzzugriffsprozess (NAP) 143  
Windows Open Systems Architecture  
  (WOSA) 143  
WOSA (Windows Open Systems  
  Architecture) 143

## X

xснаadmin (Programm) 33

## Z

Zeichen in RCF-Befehlen 129  
Zeitlimit angeben (Parameter) 104  
Zeitlimit zulassen (Parameter) 124  
Zentrale Protokollierung 37  
Ziel-TP 11, 94  
Zugeordnete Anzeige-LU (Parame-  
  ter) 116  
Zugeordnete Drucker-LU (Parame-  
  ter) 116  
Zugeordnete LUs (Parameter) 84  
Zugriff auf bestimmte LU erlauben (Para-  
  meter) 116  
Zugriff einschränken (Parameter) 100  
Zweignetz-knoten 5, 28  
Zwischenpfadsteuerung 24

---

# Antwort

IBM Communications Server für Linux  
Verwaltungshandbuch  
Version 6.2.3

IBM Form SC12-3363-02

Anregungen zur Verbesserung und Ergänzung dieser Veröffentlichung nehmen wir gerne entgegen. Bitte informieren Sie uns über Fehler, ungenaue Darstellungen oder andere Mängel.

Zur Klärung technischer Fragen sowie zu Liefermöglichkeiten und Preisen wenden Sie sich bitte entweder an Ihre IBM Geschäftsstelle, Ihren IBM Geschäftspartner oder Ihren Händler.

**Unsere Telefonauskunft "HALLO IBM" (Telefonnr.: 0180 3 313233) steht Ihnen ebenfalls zur Klärung allgemeiner Fragen zur Verfügung.**

Kommentare:

Danke für Ihre Bemühungen.

Sie können ihre Kommentare betr. dieser Veröffentlichung wie folgt senden:

- Als Brief an die Postanschrift auf der Rückseite dieses Formulars
- Als E-Mail an die folgende Adresse: [ibmterm@de.ibm.com](mailto:ibmterm@de.ibm.com)

---

Name

---

Adresse

---

Firma oder Organisation

---

Rufnummer

---

E-Mail-Adresse

IBM Deutschland GmbH  
SW TSC Germany

71083 Herrenberg





SC12-3363-02

