

IBM DB2 Information Integrator



# Konfiguration von Datenquellen

*Version 8.2*



IBM DB2 Information Integrator



# Konfiguration von Datenquellen

*Version 8.2*

**Anmerkung:**

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die allgemeinen Informationen unter „Bemerkungen“ auf Seite 721 gelesen werden.

- Die IBM Homepage finden Sie im Internet unter: **ibm.com**
- IBM und das IBM Logo sind eingetragene Marken der International Business Machines Corporation.
- Das e-business-Symbol ist eine Marke der International Business Machines Corporation.
- Infoprint ist eine eingetragene Marke der IBM.
- ActionMedia, LANDesk, MMX, Pentium und ProShare sind Marken der Intel Corporation in den USA und/oder anderen Ländern.
- C-bus ist eine Marke der Corollary, Inc. in den USA und/oder anderen Ländern.
- Java und alle auf Java basierenden Marken und Logos sind Marken der Sun Microsystems, Inc. in den USA und/oder anderen Ländern.
- Microsoft Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.
- PC Direct ist eine Marke der Ziff Communications Company in den USA und/oder anderen Ländern.
- SET und das SET-Logo sind Marken der SET Secure Electronic Transaction LLC.
- UNIX ist eine eingetragene Marke der Open Group in den USA und/oder anderen Ländern.
- Marken anderer Unternehmen/Hersteller werden anerkannt.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs

*IBM DB2 Information Integrator Data Source Configuration Guide Version 8.2,*

herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 1998, 2004

© Copyright IBM Deutschland GmbH 2004

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

Mai 2004

---

# Inhaltsverzeichnis

<b>Zu diesem Handbuch</b> . . . . .	<b>xiii</b>
Zielgruppe . . . . .	xiii
In diesem Handbuch verwendete Konventionen und Terminologie . . . . .	xiii
Lesen der Syntaxdiagramme . . . . .	xiv

---

## Teil 1. Konzepte und Planung . . . . 1

### Kapitel 1. Überblick über ein System zusammengeschlossener Datenbanken . 3

System zusammenschlossener Datenbanken . . . . .	3
Der Server mit zusammenschlossenen Datenban- ken . . . . .	4
Was ist eine Datenquelle? . . . . .	5
Unterstützte Datenquellen . . . . .	6
Die zusammenschlossene Datenbank . . . . .	8
Der Systemkatalog der zusammenschlossenen Datenbank . . . . .	9
Der SQL-Compiler . . . . .	10
Das Abfrageoptimierungsprogramm . . . . .	10
Kompensation . . . . .	11
Durchgriffssitzungen . . . . .	12
Wrapper und Wrappermodule . . . . .	13
Standardnamen für Wrapper . . . . .	14
Serverdefinitionen und Serveroptionen . . . . .	15
Benutzerzuordnungen . . . . .	16
Kurznamen und Datenquellenobjekte. . . . .	17
Gültige Datenquellenobjekte. . . . .	18
Kurznamenspaltenoptionen . . . . .	19
Datentypzuordnungen. . . . .	20
Funktionszuordnungen . . . . .	20
Indextspezifikationen . . . . .	21
Interaktion mit einem System zusammenschlosse- ner Datenbanken . . . . .	22
DB2-Befehlszeilenprozessor (CLP) . . . . .	22
DB2-Befehlszentrale . . . . .	22
DB2-Steuerzentrale . . . . .	22
Anwendungsprogramme . . . . .	23
DB2-Produktfamilie. . . . .	23
Provider für Web-Services . . . . .	24

### Kapitel 2. Planung für die Konfiguration zusammenschlossener Datenquellen 25

Namenskonventionen für Objekte in zusamme- geschlossenen Datenbanken. . . . .	25
Beibehalten der von der Groß-/Kleinschreibung abhängigen Werte in einem System zusamme- geschlossener Datenbanken . . . . .	26
Aktualisieren von Statistikdaten zur Datenquelle . . . . .	28
Auswählen des korrekten Wrappers . . . . .	29
Methoden für den Zugriff auf Excel-Daten . . . . .	30
Planen der Benutzerzuordnungen . . . . .	34
Planen der Datentypzuordnungen . . . . .	34
Planen der Funktionszuordnungen . . . . .	36

Prüfliste zur Planung der Konfiguration des Sys- tems zusammenschlossener Datenbanken . . . . .	36
Prüfliste zu den Namenskonventionen für Objekte zusammenschlossener Datenbanken . . . . .	36
Prüfliste zur Beibehaltung der von der Groß- /Kleinschreibung abhängigen Werte . . . . .	37
Prüfliste zur Datenquellenstatistik . . . . .	37
Prüfliste zu Datentypzuordnungen . . . . .	38
Prüfliste zu Benutzerzuordnungen. . . . .	38
Prüfliste für Wrapper . . . . .	39

---

## Teil 2. Server mit zusamme- geschlossenen Datenbanken und zusammenschlossene Datenbank 41

### Kapitel 3. Überprüfung der Konfigura- tion des Servers mit zusamme- geschlossenen Datenbanken . . . . . 43

Überprüfung der Konfiguration des Servers mit zusammenschlossenen Datenbanken . . . . .	43
Bestätigen der Programmverbindung (Link-edit) zu den Wrapperbibliotheksdateien (UNIX) . . . . .	43
Bestätigen der Programmverbindung (Link-edit) zu den Wrapperbibliotheksdateien (UNIX) . . . . .	43
Überprüfung der Wrapperbibliotheksdateien (UNIX) . . . . .	44
Wrapperbibliotheksdateien . . . . .	45
Überprüfung der Nachrichtendateien für das Herstellen von Programmverbindungen (UNIX) . . . . .	46
Manuelles Herstellen einer Programmverbindung zwischen den Wrapperbibliotheken und der Cli- ent-Software von Datenquellen . . . . .	47
Überprüfung des Parameters FEDERATED . . . . .	50

### Kapitel 4. Erstellen einer zusamme- geschlossenen Datenbank . . . . . 51

Überlegungen zu Landessprachen in der zusam- geschlossenen Datenbank . . . . .	51
Sortierfolgen in einem System zusammenschlosse- ner Datenbanken . . . . .	52
Einstellen der Sortierfolge einer zusammenschlos- senen Datenbank . . . . .	55
Unicode-Unterstützung für Systeme zusamme- geschlossener Datenbanken . . . . .	56
Erstellen einer zusammenschlossenen Datenbank . . . . .	58

---

## Teil 3. Datenquellen . . . . . 61

### Kapitel 5. Konfigurieren des Zugriffs auf Datenquellen - Überblick . . . . . 63

Konfigurieren von Datenquellen - Kurzüberblick . . . . .	63
--	----

Hinzufügen von Datenquellen zu einem Server mit	
zusammengeschlossenen Datenbanken mit Hilfe der	
DB2 UDB-Steuerzentrale . . . . .	64
Konfigurieren von mehreren Servern mit zusam-	
mengengeschlossenen Datenbanken für den Zugriff auf	
Datenquellen . . . . .	66
Einstellen der Umgebungsvariablen für Daten-	
quellen . . . . .	67
Festlegen der Umgebungsvariablen für eine	
Datenquelle . . . . .	67
Einschränkungen für die Datei 'db2dj.ini' . . . . .	68
Anwenden von Umgebungsvariablen in einer	
Exemplarkonfiguration mit mehreren Partitionen .	69
Registrieren von Wrappern für eine Datenquelle . . . . .	70
Registrieren von Serverdefinitionen für eine Daten-	
quelle . . . . .	71
Registrieren von Benutzerzuordnungen für eine	
Datenquelle . . . . .	72
Registrieren von Kurznamen für Datenquellen. . . . .	73
Registrieren von Kurznamen für Datenquellen. . . . .	73
Erstellen eines Kurznamens für einen Kurznamen	75
Angeben von Kurznamenspalten für eine nicht	
relationale Datenquelle . . . . .	75
Optionale Konfigurationsschritte . . . . .	76

**Kapitel 6. Konfigurieren des Zugriffs auf BioRS-Datenquellen . . . . . 77**

Was ist BioRS? . . . . .	77
Hinzufügen von BioRS zu einem Server mit zusam-	
mengengeschlossenen Datenbanken . . . . .	79
Hinzufügen von BioRS-Datenquellen zu einem	
Server mit zusammengeschlossenen Datenbanken	79
Registrieren von angepassten Funktionen für den	
BioRS-Wrapper . . . . .	80
Tabelle der angepassten Funktionen - BioRS-	
Wrapper . . . . .	81
Registrieren des BioRS-Wrappers . . . . .	81
Bibliotheksdateien für den BioRS-Wrapper . . . . .	82
Registrieren der Serverdefinition für eine BioRS-	
Datenquelle . . . . .	83
Registrieren von Benutzerzuordnungen für	
BioRS-Datenquellen. . . . .	83
Registrieren von Kurznamen für BioRS-Daten-	
quellen . . . . .	84
Anweisung CREATE NICKNAME - Beispiele für	
den BioRS-Wrapper. . . . .	86
Abfragen und angepasste Funktionen für BioRS-	
Datenquellen . . . . .	88
Angepasste Funktionen und BioRS-Abfragen . . . . .	88
Vergleichselemente für Verknüpfungen mit	
Gleichheitsattributen für den BioRS-Wrapper . . . . .	90
Das BioRS-Element 'AllText'. . . . .	92
BioRS-Wrapper - Beispielabfragen . . . . .	93
Optimieren der Leistung des BioRS-Wrappers . . . . .	98
Richtlinien zur Leistungsoptimierung des BioRS-	
Wrappers . . . . .	98
Statistische BioRS-Daten . . . . .	99
Ermitteln der Kardinalitätsstatistiken von BioRS-	
Datenbanken . . . . .	100
Aktualisieren der Kardinalitätsstatistiken von	
BioRS-Kurznamen . . . . .	100

Aktualisieren der Kardinalitätsstatistiken von	
BioRS-Spalten . . . . .	101
Aktualisieren der Kardinalität der BioRS-Spalte	
_ID_ . . . . .	103
Nachrichten für den BioRS-Wrapper. . . . .	104

**Kapitel 7. Konfigurieren des Zugriffs auf BLAST-Datenquellen . . . . . 109**

Was ist BLAST? . . . . .	109
Hinzufügen von BLAST zu einem Server mit	
zusammengeschlossenen Datenbanken . . . . .	113
Hinzufügen von BLAST-Datenquellen zu einem	
Server mit zusammengeschlossenen Datenban-	
ken . . . . .	113
Überprüfen, ob die korrekte Version der aus-	
führbaren Datei 'blastall' und der Matrixdateien	
installiert wurde . . . . .	114
Konfigurieren des BLAST-Dämons . . . . .	115
Konfigurationsdatei für den BLAST-Dämon -	
Beispiele . . . . .	118
Starten des BLAST-Dämons. . . . .	119
Befehl db2blast_daemon - Syntax und Beispiele	120
Registrieren des BLAST-Wrappers . . . . .	121
Bibliotheksdateien des BLAST-Wrappers . . . . .	122
Registrieren des Servers für BLAST-Daten-	
quellen . . . . .	122
Registrieren von Kurznamen für BLAST-Daten-	
quellen . . . . .	123
Einrichten von TurboBlast zum Arbeiten mit dem	
BLAST-Wrapper . . . . .	130
Aufbau von BLAST-SQL-Abfragen . . . . .	131
BLAST-Datenquelle - Beispielabfragen . . . . .	132
Tipps für die Optimierung des BLAST-Wrappers	134
Nachrichten für den BLAST-Wrapper . . . . .	134

**Kapitel 8. Konfigurieren des Zugriffs auf Geschäftsanwendungsdaten- quellen . . . . . 137**

Der WebSphere Business Integration-Wrapper . . . . .	137
Geschäftsobjektdefinitionen. . . . .	139
Konfigurieren des WebSphere Business Integration-	
Adapters . . . . .	141
Hinzufügen von Geschäftsanwendungen zu einem	
Server mit zusammengeschlossenen Datenbanken .	144
Hinzufügen von Geschäftsanwendungsdaten-	
quellen zu einem System zusammengeschlosse-	
ner Datenbanken . . . . .	144
Registrieren des WebSphere Business Integri-	
tion-Wrappers . . . . .	145
Bibliotheksdateien des WebSphere Business Inte-	
gration-Wrappers . . . . .	146
Registrieren der Serverdefinition für Geschäfts-	
anwendungsdatenquellen . . . . .	146
Registrieren von Kurznamen für Geschäfts-	
anwendungsdatenquellen . . . . .	148
Einschränkungen bei Abfragen für Wrapper für	
Geschäftsanwendungen und Web-Services. . . . .	172
Geschäftsanwendungsdatenquellen – Beispielab-	
fragen . . . . .	175

## Kapitel 9. Konfigurieren des Zugriffs auf Datenquellen der DB2-Produktfamilie . . . . . 179

Hinzufügen von Datenquellen der DB2-Produktfamilie zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	179
Hinzufügen von Datenquellen der DB2-Produktfamilie zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	179
Katalogisieren eines Knoteneintrags in das zusammengeschlossene Knotenverzeichnis . . . . .	180
Katalogisieren der fernen Datenbank im Datenbankverzeichnis des Systems zusammengeschlossener Datenbanken. . . . .	181
Registrieren des DB2-Wrappers . . . . .	182
Bibliotheksdateien des DB2-Wrappers . . . . .	183
Registrieren der Serverdefinitionen für eine DB2-Datenquelle . . . . .	184
Anweisung CREATE SERVER - Beispiele für den DB2-Wrapper . . . . .	185
Erstellen der Benutzerzuordnung für eine DB2-Datenquelle . . . . .	186
Anweisung CREATE USER MAPPING - Beispiele für den DB2-Wrapper . . . . .	187
Testen der Verbindung zum DB2-Datenquellen-server . . . . .	188
Registrieren von Kurznamen für DB2-Tabellen und -Sichten. . . . .	189
Anweisung CREATE NICKNAME - Beispiele für den DB2-Wrapper . . . . .	190

## Kapitel 10. Konfigurieren des Zugriffs auf Documentum-Datenquellen. . . . . 191

Was ist Documentum? . . . . .	191
Hinzufügen von Documentum zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	193
Hinzufügen von Documentum-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	193
Bereitstellen der Documentum-Clientbibliothek für den Wrapper . . . . .	194
Festlegen der Documentum-Umgebungsvariablen . . . . .	195
Registrieren des Documentum-Wrappers . . . . .	196
Bibliotheksdateien für den Documentum-Wrapper . . . . .	197
Registrieren des Servers für Documentum-Datenquellen . . . . .	198
Registrieren der Benutzerzuordnungen für Documentum-Datenquellen. . . . .	198
Registrieren von Kurznamen für Documentum-Datenquellen . . . . .	199
Abfragen und angepasste Funktionen für Documentum-Datenquellen . . . . .	206
Registrieren der angepassten Funktionen für den Documentum-Wrapper. . . . .	206
Documentum-Datenquelle - Beispielabfragen	212
Zugriffssteuerung für den Documentum-Wrapper . . . . .	214
Nachrichten für den Documentum-Wrapper . . . . .	214

## Kapitel 11. Konfigurieren des Zugriffs auf Entrez-Datenquellen . . . . . 221

Was ist Entrez? . . . . .	221
Hinzufügen von Entrez zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	222
Hinzufügen von Entrez-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken. . . . .	222
Registrieren von angepassten Funktionen für den Entrez-Wrapper . . . . .	223
Angepasste Funktionstabelle - Entrez-Wrapper	224
Registrieren des Entrez-Wrappers. . . . .	225
Bibliotheksdateien des Entrez-Wrappers . . . . .	225
Registrieren des Servers für eine Entrez-Datenquelle . . . . .	226
Registrieren von Kurznamen für Entrez-Datenquellen . . . . .	228
Abfragen und angepasste Funktionen für Entrez-Datenquellen . . . . .	230
Angepasste Funktionen und Entrez-Abfragen	230
Relationale Vergleichselemente für den Entrez-Wrapper . . . . .	232
Ungültige WHERE-Klauseln für den Entrez-Wrapper . . . . .	232
Vereinfachung von Schemadatenelementen . . . . .	232
Entrez-Datenquelle - Beispielabfragen . . . . .	233
PubMed-Schematabellen. . . . .	235
Nucleotide-Schematabellen . . . . .	240
Nachrichten für den Entrez-Wrapper . . . . .	244

## Kapitel 12. Konfigurieren des Zugriffs auf Excel-Datenquellen . . . . . 249

Was ist Excel? . . . . .	249
Hinzufügen von Excel zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	250
Hinzufügen von Excel-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken. . . . .	250
Registrieren des Excel-Wrappers . . . . .	251
Bibliotheksdateien für den Excel-Wrapper . . . . .	252
Registrieren des Servers für eine Excel-Datenquelle . . . . .	253
Registrieren von Kurznamen für Excel-Datenquellen . . . . .	253
Excel-Datenquelle - Beispielabfragen. . . . .	254
Excel-Datenquelle - Beispielszenario . . . . .	255
Modell für die Dateizugriffssteuerung für den Excel-Wrapper . . . . .	256
Nachrichten für den Excel-Wrapper . . . . .	257

## Kapitel 13. Konfigurieren des Zugriffs auf Extended Search-Datenquellen . . . . . 263

Was ist Extended Search? . . . . .	263
Extended Search-Datenquellen. . . . .	264
Funktionsweise des Extended Search-Wrappers	265
Kurznamen für Extended Search . . . . .	266
Vertikale Tabellen für Extended Search . . . . .	267
Hinzufügen von Extended Search zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	270

Hinzufügen von Extended Search-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	270
Registrieren des Extended Search-Wrappers . . . . .	271
Bibliotheksdateien für den Extended Search-Wrapper . . . . .	272
Registrieren des Servers für Extended Search-Datenquellen . . . . .	272
Registrieren von Benutzerzuordnungen für Extended Search-Datenquellen. . . . .	273
Registrieren von Kurznamen für Extended Search-Datenquellen . . . . .	274
Registrieren der angepassten Funktionen für den Extended Search-Wrapper. . . . .	275
Abfragen von Extended Search-Datenquellen. . . . .	276
Extended Search-Wrapper - Richtlinien für Abfragen . . . . .	276
Extended Search-Wrapper - Beispielabfragen	278
Extended Search-Wrapper - Abfragesprache GQL (Generalized Query Language). . . . .	281
Nachrichten für den Extended Search-Wrapper . . . . .	283

## Kapitel 14. Konfigurieren des Zugriffs auf HMMER-Datenquellen . . . . . 287

Was ist HMMER? . . . . .	287
Hinzufügen von HMMER zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	290
Hinzufügen von HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	290
Überprüfen der Version der ausführbaren HMMER-Programmdatei . . . . .	291
Konfigurieren des HMMER-Dämons . . . . .	291
Konfigurationsdatei für den HMMER-Dämon - Beispiele . . . . .	295
Starten des HMMER-Dämons . . . . .	296
Befehl db2hmmmer_daemon - Syntax und Beispiele . . . . .	297
Registrieren des HMMER-Wrappers . . . . .	299
Bibliotheksdateien für den HMMER-Wrapper	299
Registrieren der Serverdefinition für eine HMMER-Datenquelle. . . . .	300
Anweisung CREATE SERVER - Beispiele für HMMER-Wrapper . . . . .	300
Registrieren von Kurznamen für HMMER-Datenquellen . . . . .	302
Fixierte Spalten für HMMER-Kurznamen . . . . .	303
Anweisung CREATE NICKNAME - Beispiel für den HMMER-Wrapper . . . . .	307
HMMER-Datenquelle - vollständiges Beispiel	308
Erstellen neuer HMMER-Abfragen mit Hilfe von Beispielen . . . . .	309
Nachrichten für den HMMER-Wrapper. . . . .	311

## Kapitel 15. Konfigurieren des Zugriffs auf Informix-Datenquellen . . . . . 313

Hinzufügen von Informix zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	313
Hinzufügen von Informix-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken. 313	

Einrichten und Testen der Informix-Clientkonfigurationsdatei . . . . .	314
Festlegen der Informix-Umgebungsvariablen	315
Registrieren des Informix-Wrappers . . . . .	319
Bibliotheksdateien für den Informix-Wrapper	320
Registrieren der Serverdefinitionen für eine Informix-Datenquelle. . . . .	321
Anweisung CREATE SERVER - Beispiele für den Informix-Wrapper . . . . .	322
Erstellen der Benutzerzuordnung für eine Informix-Datenquelle . . . . .	323
Anweisung CREATE USER MAPPING - Beispiel für den Informix-Wrapper . . . . .	324
Testen der Verbindung zum Informix-Server . . . . .	325
Registrieren von Kurznamen für Informix-Tabellen, -Sichten und -Synonyme . . . . .	326
Anweisung CREATE NICKNAME - Beispiele für den Informix-Wrapper . . . . .	327
Optimierung und Fehlerbehebung der Konfiguration für Informix-Datenquellen . . . . .	328
Verbessern der Leistung durch Definieren der Serveroptionen FOLD_ID und FOLD_PW . . . . .	328

## Kapitel 16. Konfigurieren des Zugriffs auf Microsoft SQL Server-Datenquellen . . . . . 331

Hinzufügen von Microsoft SQL Server zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	331
Hinzufügen von Microsoft SQL Server-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	331
Vorbereiten des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf Microsoft SQL Server-Datenquellen . . . . .	332
Festlegen der Umgebungsvariablen für Microsoft SQL Server. . . . .	334
Registrieren des Microsoft SQL Server-Wrappers	336
Bibliotheksdateien des Microsoft SQL Server-Wrappers. . . . .	337
Registrieren der Serverdefinitionen für eine Microsoft SQL Server-Datenquelle. . . . .	338
Anweisung CREATE SERVER - Beispiele für den Microsoft SQL Server-Wrapper . . . . .	339
Erstellen einer Benutzerzuordnung für eine Microsoft SQL Server-Datenquelle. . . . .	341
Anweisung CREATE USER MAPPING - Beispiele für den Microsoft SQL Server-Wrapper. . . . .	341
Testen der Verbindung zum fernen Server mit Microsoft SQL Server. . . . .	342
Registrieren von Kurznamen für Microsoft SQL Server-Tabellen und Sichten . . . . .	344
Anweisung CREATE NICKNAME - Beispiele für den Microsoft SQL Server-Wrapper. . . . .	345
Optimieren der Konfiguration für Microsoft SQL Server-Datenquellen und Beheben von Konfigurationsfehlern . . . . .	345
Abrufen von ODBC-Traces . . . . .	345

## Kapitel 17. Konfigurieren des Zugriffs auf ODBC-Datenquellen . . . . . 347



Hinzufügen von ODBC zu einem System zusammenge- schlossener Datenbanken . . . . .	347	Einrichten und Testen der Oracle-Clientkonfigu- rationsdatei . . . . .	376
Hinzufügen von ODBC-Datenquellen zu einem Server mit zusammengeschlossenen Datenban- ken. . . . .	347	Registrieren des Oracle-Wrappers. . . . .	377
Vorbereiten des Servers mit zusammengeschlos- senen Datenbanken für den Zugriff auf Daten- quellen über ODBC . . . . .	349	Oracle-Wrapper-Bibliotheksdateien . . . . .	378
Registrieren des ODBC-Wrappers. . . . .	350	Registrieren der Serverdefinitionen für eine Ora- cle-Datenquelle. . . . .	378
ODBC-Wrapper-Bibliotheksdateien . . . . .	351	Anweisung CREATE SERVER - Beispiele für den Oracle-Wrapper . . . . .	379
Anweisung CREATE WRAPPER - Beispiele für den ODBC-Wrapper . . . . .	351	Erstellen der Benutzerzuordnungen für eine Oracle-Datenquelle . . . . .	381
Registrieren der Serverdefinitionen für eine ODBC-Datenquelle . . . . .	352	Anweisung CREATE USER MAPPING - Bei- spiele für den Oracle-Wrapper. . . . .	381
Anweisung CREATE SERVER - Beispiele für den ODBC-Wrapper . . . . .	353	Testen der Verbindung zum Oracle-Server. . . . .	382
Erstellen einer Benutzerzuordnung für eine ODBC-Datenquelle . . . . .	354	Registrieren von Kurznamen für Oracle-Tabellen und -Sichten. . . . .	384
Anweisung CREATE USER MAPPING - Bei- spiele für den ODBC-Wrapper. . . . .	355	Anweisung CREATE NICKNAME - Beispiele für den Oracle-Wrapper . . . . .	385
Testen der Verbindung zum ODBC-Daten- quellenserver . . . . .	356	Optimierung und Fehlerbehebung Konfiguration für Oracle-Datenquellen . . . . .	386
Registrieren von Kurznamen für ODBC-Daten- quellentabellen und -sichten . . . . .	357	Konnektivitätsprobleme . . . . .	386
Anweisung CREATE NICKNAME - Beispiele für den ODBC-Wrapper . . . . .	358	<b>Kapitel 20. Konfigurieren des Zugriffs auf Sybase-Datenquellen . . . . .</b>	<b>387</b>
Zugreifen auf Excel-Daten mit dem ODBC-Wrap- per. . . . .	359	Hinzufügen von Sybase zu einem System zusammen- geschlossener Datenbanken . . . . .	387
Ändern der standardmäßigen Datentypzu- ordnungen . . . . .	361	Hinzufügen von Sybase-Datenquellen zu einem Server mit zusammengeschlossenen Datenban- ken. . . . .	387
Anweisung ALTER SERVER - Beispiele für den ODBC-Wrapper für den Zugriff auf Excel-Daten. . . . .	362	Einstellen der Sybase-Umgebungsvariablen . . . . .	388
Optimieren der Konfiguration für ODBC-Daten- quellen und Beheben von Konfigurationsfehlern. . . . .	363	Einrichten und Testen der Sybase-Clientkonfigu- rationsdatei . . . . .	391
Abrufen von ODBC-Traces . . . . .	363	Registrieren des Sybase-Wrappers . . . . .	392
<b>Kapitel 18. Konfigurieren des Zugriffs auf OLE DB-Datenquellen . . . . .</b>	<b>365</b>	Sybase-Wrapper-Bibliotheksdateien . . . . .	393
Hinzufügen von OLE DB-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	365	Registrieren der Serverdefinitionen für eine Sybase-Datenquelle . . . . .	394
Registrieren des OLE DB-Wrappers . . . . .	366	Anweisung CREATE SERVER - Beispiele für den Sybase-Wrapper . . . . .	395
OLE DB-Wrapper-Bibliotheksdateien . . . . .	367	Erstellen einer Benutzerzuordnung für eine Sybase-Datenquelle . . . . .	397
Registrieren der Serverdefinitionen für eine OLE DB-Datenquelle. . . . .	367	Anweisung CREATE USER MAPPING - Bei- spiele für den Sybase-Wrapper . . . . .	398
Anweisung CREATE SERVER - Beispiele für den OLE DB-Wrapper . . . . .	368	Testen der Verbindung zum Sybase-Server. . . . .	399
Erstellen einer Benutzerzuordnung für eine OLE DB-Datenquelle. . . . .	369	Registrieren von Kurznamen für Sybase-Tabel- len und -Sichten . . . . .	400
Anweisung CREATE USER MAPPING - Beispiele für den OLE DB-Wrapper . . . . .	370	Anweisung CREATE NICKNAME - Beispiele für den Sybase-Wrapper. . . . .	401
<b>Kapitel 19. Konfigurieren des Zugriffs auf Oracle-Datenquellen. . . . .</b>	<b>371</b>	Optimierung und Fehlerbehebung Konfiguration für Sybase-Datenquellen. . . . .	402
Hinzufügen von Oracle zu einem System zusammen- geschlossener Datenbanken . . . . .	371	Auflösen des Fehlers sp_helpindex . . . . .	402
Hinzufügen von Oracle-Datenquellen zu einem Server mit zusammengeschlossenen Datenban- ken. . . . .	371	<b>Kapitel 21. Konfigurieren des Zugriffs auf Datenquellen in Dateien mit Tabellenstruktur . . . . .</b>	<b>403</b>
Einrichten der Oracle-Umgebungsvariablen . . . . .	372	Was sind Dateien mit Tabellenstruktur? . . . . .	403
		Attribute von Dateien mit Tabellenstruktur . . . . .	404
		Sortierte Dateien . . . . .	404
		Unsortierte Dateien . . . . .	404
		Arbeiten mit Dateien mit Tabellenstruktur in DB2	
		Information Integrator . . . . .	404

Hinzufügen von Dateien mit Tabellenstruktur zu einem System zusammengeschlossener Datenbanken . . . . .	406
Hinzufügen von Datenquellen von Dateien mit Tabellenstruktur zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	406
Registrieren des Wrappers für Dateien mit Tabellenstruktur . . . . .	407
Bibliotheksdateien des Wrappers für Dateien mit Tabellenstruktur . . . . .	407
Registrieren des Servers für Dateien mit Tabellenstruktur . . . . .	408
Registrieren der Kurznamen für Dateien mit Tabellenstruktur . . . . .	408
Anweisung CREATE NICKNAME - Beispiele für den Wrapper für Dateien mit Tabellenstruktur . . . . .	409
Modell der Dateizugriffssteuerung für den Wrapper für Dateien mit Tabellenstruktur . . . . .	410
Tipps und Überlegungen für die Optimierung des Wrappers für Dateien mit Tabellenstruktur . . . . .	411
Nachrichten für den Wrapper für Dateien mit Tabellenstruktur . . . . .	411

**Kapitel 22. Konfigurieren des Zugriffs auf Teradata-Datenquellen . . . . . 415**

Hinzufügen von Teradata zu einem System zusammengeschlossener Datenbanken . . . . .	415
Hinzufügen von Teradata-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	415
Testen der Verbindung zum Teradata-Server . . . . .	416
Prüfen, ob die Teradata-Bibliothek für das Binden zur Laufzeit aktiviert ist (AIX) . . . . .	417
Einstellen der Teradata-Umgebungsvariablen . . . . .	418
Registrieren des Teradata-Wrappers . . . . .	421
Teradata-Wrapper-Bibliotheksdateien . . . . .	422
Registrieren der Serverdefinitionen für eine Teradata-Datenquelle . . . . .	422
Anweisung CREATE SERVER - Beispiele für Teradata-Wrapper . . . . .	424
Erstellen der Benutzerzuordnung für eine Teradata-Datenquelle . . . . .	425
Anweisung CREATE USER MAPPING - Beispiele für Teradata-Wrapper . . . . .	425
Testen der Verbindung vom Server mit zusammengeschlossenen Datenbanken zum Teradata-Server . . . . .	426
Teradata-Kurznamen auf Servern mit zusammengeschlossenen Datenbanken . . . . .	427
Registrieren von Kurznamen für Teradata-Tabellen und -Sichten . . . . .	428
Anweisung CREATE NICKNAME - Beispiele für den Teradata-Wrapper . . . . .	429
Optimierung und Fehlerbehebung der Konfiguration für Teradata-Datenquellen . . . . .	430
Fehler bei UPDATE- oder DELETE-Operationen für Kurznamen . . . . .	430
Optimierung und Inaktivierung der Teradata-Zugriffsprotokollierung . . . . .	430

Aktivieren des Bindens zur Laufzeit für libcliv2.so (AIX) . . . . .	430
---	-----

**Kapitel 23. Konfigurieren des Zugriffs auf Web-Services-Datenquellen . . . . . 433**

Der Web-Services-Wrapper und das WSDL-Dokument . . . . .	433
Hinzufügen von Web-Services zu einem System zusammengeschlossener Datenbanken . . . . .	439
Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken . . . . .	439
Registrieren des Web-Services-Wrappers . . . . .	440
Bibliotheksdateien des Web-Services-Wrappers . . . . .	441
Registrieren der Serverdefinition für Web-Services-Datenquellen . . . . .	441
Registrieren von Kurznamen für Web-Services-Datenquellen . . . . .	442
Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services . . . . .	459
Web-Services-Datenquellen – Beispielabfragen . . . . .	463
Web-Services-Wrapper - Sicherheit . . . . .	467
Nachrichten für den Web-Services-Wrapper . . . . .	468

**Kapitel 24. Konfigurieren des Zugriffs auf XML-Datenquellen. . . . . 471**

Was ist XML? . . . . .	471
Hinzufügen von XML zu einem System zusammengeschlossener Datenbanken . . . . .	474
Hinzufügen von XML zu einem System zusammengeschlossener Datenbanken . . . . .	474
Registrieren des XML-Wrappers . . . . .	475
XML-Wrapper-Bibliotheksdateien . . . . .	476
Registrieren des Servers für XML-Datenquellen . . . . .	477
Registrieren von Kurznamen für XML-Datenquellen . . . . .	479
Erstellen von Sichten für zusammengeschlossene Datenbanken für Nicht-Stammkurznamen (XML-Wrapper) . . . . .	487
Optimierungstipps für die XML-Aufwandmodellfunktion . . . . .	488
XML-Datenquelle - Beispielabfragen . . . . .	489
Nachrichten für den XML-Wrapper . . . . .	491

**Teil 4. Benutzerdefinierte Funktionen . . . . . 501**

**Kapitel 25. Benutzerdefinierte Life Sciences-Funktionen . . . . . 503**

Benutzerdefinierte Life Sciences-Funktionen - Übersicht . . . . .	503
Benutzerdefinierte Life Sciences-Funktionen - Übersicht . . . . .	503
Bibliotheksdateien der benutzerdefinierten Life Sciences-Funktionen . . . . .	503
Benutzerdefinierte Life Sciences-Funktionen nach Funktionskategorie . . . . .	504
Registrieren von benutzerdefinierten Life Sciences-Funktionen . . . . .	505

Inaktivieren der benutzerdefinierten Life Sciences-Funktionen . . . . .	507	Benutzerdefinierte Funktion 'LSTransAllFrames' - Beispiel . . . . .	540
Benutzerdefinierte Rückübersetzungsfunktionen	508	Format der Codon-Frequenztafel . . . . .	541
Benutzerdefinierte Rückübersetzungsfunktionen - Übersicht . . . . .	508	Codon-Frequenztafel - Beispiel . . . . .	542
Benutzerdefinierte Funktion 'LSPep2AmbNuc'	508	Format von Übersetzungstabellen . . . . .	542
Benutzerdefinierte Funktion 'LSPep2AmbNuc' - Beispiel . . . . .	510	Übersetzungstabelle - Beispiel . . . . .	543
Benutzerdefinierte Funktion 'LSPep2AmbNuc' - Fehlernachrichten . . . . .	511	<b>Kapitel 26. Benutzerdefinierte KEGG-Funktionen . . . . . 545</b>	
Benutzerdefinierte Funktion 'LSPep2ProbNuc'	511	Benutzerdefinierte KEGG-Funktionen - Übersicht	545
Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Beispiel . . . . .	512	Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie . . . . .	546
Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Fehlernachrichten . . . . .	513	Funktionsargumente für benutzerdefinierte KEGG-Funktionen . . . . .	547
Benutzerdefinierte Funktionen für die Syntaxanalyse von Definitionszeilen . . . . .	514	Registrieren der benutzerdefinierten KEGG-Funktionen . . . . .	548
Benutzerdefinierte Funktion für die Syntaxanalyse von Definitionszeilen - Übersicht . . . . .	514	Pathway-Datenbankfunktionen . . . . .	550
Benutzerdefinierte LSDefineParse-Funktionen	514	Benutzerdefinierte Funktion GenesByPathwyS	550
Benutzerdefinierte Funktionen 'LSDefineParse' - Beispiele . . . . .	517	Benutzerdefinierte Funktion GenesByPathwyT	551
Benutzerdefinierte Funktionen für allgemeine Mustererkennung . . . . .	521	Benutzerdefinierte Funktion CompoundsByPathwyS . . . . .	552
Benutzerdefinierte Funktion 'LSPatternMatch'	521	Benutzerdefinierte Funktion CompoundsByPathwyT . . . . .	552
Benutzerdefinierte Funktion 'LSPatternMatch' - Beispiel . . . . .	521	Benutzerdefinierte Funktion EnzymesByPathwyS . . . . .	553
Benutzerdefinierte Funktion 'LSPrositePattern'	523	Benutzerdefinierte Funktion EnzymesByPathwyT . . . . .	554
Benutzerdefinierte Funktion 'LSPrositePattern' - Beispiel . . . . .	524	Benutzerdefinierte Funktion PathwysByGenesS	555
Unterstützung für reguläre Ausdrücke . . . . .	524	Benutzerdefinierte Funktion PathwysByGenesT	556
Benutzerdefinierte Funktion 'GeneWise' . . . . .	525	Benutzerdefinierte Funktion PathwysByCompndsS . . . . .	557
Verbindung zu 'GeneWise' herstellen . . . . .	525	Benutzerdefinierte Funktion PathwysByCompndsT . . . . .	558
Benutzerdefinierte Funktion 'LSGeneWise' . . . . .	526	Benutzerdefinierte Funktion PathwysByEnzymesS . . . . .	558
Benutzerdefinierte Funktion 'LSGeneWise' - Beispiel . . . . .	527	Benutzerdefinierte Funktion PathwysByEnzymesT . . . . .	559
Benutzerdefinierte Motivfunktionen . . . . .	528	Sequence Similarity-Datenbankfunktionen . . . . .	560
Benutzerdefinierte Funktion 'LSBarCode' . . . . .	528	Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten . . . . .	560
Benutzerdefinierte Funktion 'LSBarCode' - Beispiel . . . . .	528	Benutzerdefinierte Funktion AllNbrsByGeneS	561
Benutzerdefinierte Funktion 'LSMultiMatch' . . . . .	529	Benutzerdefinierte Funktion AllNbrsByGeneT	562
Benutzerdefinierte Funktion 'LSMultiMatch' - Beispiel . . . . .	530	Benutzerdefinierte Funktion BstBstNbrsByGeneS	564
Benutzerdefinierte Funktion 'LSMultiMatch3'	531	Benutzerdefinierte Funktion BstBstNbrsByGeneT	565
Benutzerdefinierte Funktion 'LSMultiMatch3' - Beispiel . . . . .	531	Benutzerdefinierte Funktion BestNbrsByGeneS	566
Benutzerdefinierte Umkehrfunktionen . . . . .	533	Benutzerdefinierte Funktion BestNbrsByGeneT	567
Benutzerdefinierte Funktion 'LSRevComp' . . . . .	533	Benutzerdefinierte Funktion RevBestNbrsByGeneS . . . . .	568
Benutzerdefinierte Funktion 'LSRevComp' - Beispiel . . . . .	534	Benutzerdefinierte Funktion RevBestNbrsByGeneT . . . . .	569
Benutzerdefinierte Funktion 'LSRevNuc' . . . . .	535	Benutzerdefinierte Funktion ParalogsByGeneS	570
Benutzerdefinierte Funktion 'LSRevNuc' - Beispiel . . . . .	535	Benutzerdefinierte Funktion ParalogsByGeneT	571
Benutzerdefinierte Funktion 'LSRevPep' . . . . .	536	Benutzerdefinierte Funktion BestHmlgsByGeneS . . . . .	572
Benutzerdefinierte Funktion 'LSRevPep' - Beispiel . . . . .	536	Benutzerdefinierte Funktion BestHmlgsByGeneT . . . . .	573
Benutzerdefinierte Übersetzungsfunktionen . . . . .	537	Benutzerdefinierte Funktion BstBstHmlgByGeneS . . . . .	573
Benutzerdefinierte Funktion 'LSNuc2Pep' . . . . .	537	Benutzerdefinierte Funktion BstBstHmlgByGeneT . . . . .	574
Benutzerdefinierte Funktion 'LSNuc2Pep' - Beispiel . . . . .	538		
Benutzerdefinierte Funktion 'LSTransAllFrames'	539		

Benutzerdefinierte Funktion ScoreBetweenGenesS . . . . .	575
Benutzerdefinierte Funktion DefinitionsByGenes	576
Benutzerdefinierte Funktion GenesByMotifsT	577
Inaktivieren der benutzerdefinierten KEGG-Funktionen . . . . .	577

**Teil 5. Referenz . . . . . 579**

**Kapitel 27. Unterstützte Datentypen für nicht relationale Datenquellen. . . 581**

Unterstützte Datentypen für nicht relationale Datenquellen . . . . .	581
Vom BioRS-Wrapper unterstützte Datentypen	581
Vom BLAST-Wrapper unterstützte Datentypen	581
Vom Documentum-Wrapper unterstützte Datentypen . . . . .	582
Vom Entrez-Wrapper unterstützte Datentypen	582
Vom Excel-Wrapper unterstützte Datentypen	583
Vom Extended Search-Wrapper unterstützte Datentypen . . . . .	583
Vom HMMER-Wrapper unterstützte Datentypen	583
Vom Wrapper für Dateien mit Tabellenstruktur unterstützte Datentypen . . . . .	584
Vom Web-Services-Wrapper unterstützte Datentypen . . . . .	584
Vom WebSphere Business Integration-Wrapper unterstützte Datentypen . . . . .	585
Vom XML-Wrapper unterstützte Datentypen	585

**Kapitel 28. Ändern von Kurznamen 587**

Ändern eines Kurznamens . . . . .	587
Einschränkungen für das Ändern von Kurznamen	589
Ändern der Spaltennamen von Kurznamen . . . . .	590
Ändern von Kurznamenoptionen. . . . .	592
Ändern der Spaltenoptionen von Kurznamen . . . . .	593
Ändern eines lokalen Typs für ein Datenquellenobjekt . . . . .	595
Ändern eines lokalen Typs für ein Datenquellenobjekt - Beispiele . . . . .	597

**Kapitel 29. DDL-Befehlsreferenz . . . 599**

BioRS-DDL-Referenzinformationen . . . . .	599
Anweisungsoptionen für CREATE SERVER - BioRS-Wrapper . . . . .	599
Anweisungsoptionen für CREATE USER MAPPING - BioRS-Wrapper . . . . .	600
Anweisungssyntax für CREATE NICKNAME - BioRS-Wrapper . . . . .	601
BLAST-DDL-Referenzinformationen . . . . .	602
CREATE SERVER-Anweisungsargumente - BLAST-Wrapper . . . . .	602
CREATE NICKNAME-Anweisungssyntax - BLAST-Wrapper . . . . .	603
Documentum-DDL-Referenzinformationen . . . . .	604
CREATE SERVER-Anweisungsargumente und -optionen - Documentum-Wrapper . . . . .	604
CREATE USER MAPPING-Anweisungsoptionen - Documentum-Wrapper. . . . .	605

CREATE NICKNAME-Anweisungssyntax - Documentum-Wrapper . . . . .	606
Excel-DDL-Referenzinformationen . . . . .	609
CREATE SERVER-Anweisungsargumente - Excel-Wrapper . . . . .	609
CREATE NICKNAME-Anweisungssyntax - Excel-Wrapper . . . . .	609
Extended Search-DDL-Referenzinformationen . . . . .	611
CREATE WRAPPER-Anweisungssyntax - Extended Search-Wrapper . . . . .	611
CREATE SERVER-Anweisungssyntax - Extended Search-Wrapper. . . . .	611
CREATE USER MAPPING-Anweisungssyntax - Extended Search-Wrapper . . . . .	613
CREATE NICKNAME-Anweisungssyntax - Extended Search-Wrapper . . . . .	613
CREATE FUNCTION-Anweisungssyntax - Extended Search-Wrapper . . . . .	616
Entrez-DDL-Referenzinformationen . . . . .	617
CREATE SERVER-Anweisungsargumente - Entrez-Wrapper . . . . .	617
CREATE NICKNAME-Anweisungsoptionen - Entrez-Wrapper . . . . .	617
Referenzinformationen zur DDL von Dateien mit Tabellenstruktur . . . . .	618
Syntax der Anweisung CREATE NICKNAME - Wrapper für Dateien mit Tabellenstruktur . . . . .	618
XML-DDL-Referenzinformationen . . . . .	623
Syntax der Anweisung CREATE NICKNAME - XML-Wrapper . . . . .	623

**Anhang A. Sichten in der globalen Katalogtabelle mit Informationen der zusammengeschlossenen Datenbank . 629**

**Anhang B. Wrapperoptionen für Systeme zusammengeschlossener Datenbanken . . . . . 633**

**Anhang C. Gültige Servertypen in SQL-Anweisungen . . . . . 635**

BioRS-Wrapper . . . . .	635
BLAST-Wrapper . . . . .	635
CTLIB-Wrapper . . . . .	636
Documentum-Wrapper . . . . .	636
DRDA-Wrapper . . . . .	636
Entrez-Wrapper . . . . .	637
Excel-Wrapper . . . . .	637
Extended Search-Wrapper . . . . .	637
HMMER-Wrapper . . . . .	637
Informix-Wrapper . . . . .	638
MSSQLODBC3-Wrapper. . . . .	638
NET8-Wrapper . . . . .	638
ODBC-Wrapper . . . . .	638
OLE DB-Wrapper . . . . .	638
Wrapper für Dateien mit Tabellenstruktur . . . . .	638
Teradata-Wrapper . . . . .	639
Wrapper für Web-Services . . . . .	639
WebSphere Business Integration-Wrapper . . . . .	639



XML-Wrapper . . . . .	639
<b>  Anhang D. Serveroptionen für Systeme zusammengeschnittener Datenbanken . . . . .</b>	<b>641</b>
<b>  Anhang E. Benutzerzuordnungsoptionen für Systeme zusammengeschnittener Datenbanken . . . . .</b>	<b>659</b>
<b>  Anhang F. Kurznamenoptionen für Systeme zusammengeschnittener Datenbanken . . . . .</b>	<b>661</b>
<b>  Anhang G. Kurznamenspaltenoptionen für Systeme zusammengeschnittener Datenbanken . . . . .</b>	<b>671</b>
<b>  Anhang H. Standardmäßige progressive Datentypzuordnungen . . . . .</b>	<b>679</b>
Datenquellen von DB2 für z/OS und OS/390 . . . . .	679
Datenquellen von DB2 für iSeries. . . . .	680
Datenquellen von DB2 Server für VM und VSE . . . . .	682
Datenquellen von DB2 für Linux, UNIX und Windows . . . . .	683
Informix-Datenquellen . . . . .	684
Microsoft SQL Server-Datenquellen . . . . .	685
ODBC-Datenquellen . . . . .	688
Oracle NET8-Datenquellen . . . . .	689
Sybase-Datenquellen . . . . .	690
Teradata-Datenquellen . . . . .	691
<b>  Anhang I. Standardmäßige regressive Typzuordnungen . . . . .</b>	<b>695</b>
Datenquellen von DB2 für z/OS und OS/390 . . . . .	696
Datenquellen von DB2 für iSeries. . . . .	697
Datenquellen von DB2 für VM und VSE . . . . .	698
Datenquellen von DB2 für Linux, UNIX und Windows . . . . .	699
Informix-Datenquellen . . . . .	700
Microsoft SQL Server-Datenquellen . . . . .	701
Oracle NET8-Datenquellen . . . . .	702
Sybase-Datenquellen . . . . .	703
Teradata-Datenquellen . . . . .	704
<b>  Anhang J. Funktionszuordnungsoptionen für Systeme zusammengeschnittener Datenbanken . . . . .</b>	<b>707</b>

<b>DB2 Information Integrator-Dokumentation . . . . .</b>	<b>709</b>
Zugriff auf die DB2 Information Integrator-Dokumentation . . . . .	709
Dokumentation zur Replikationsfunktion unter z/OS . . . . .	711
Dokumentation zur Event-Publishing-Funktion für DB2 Universal Database unter z/OS. . . . .	712
Dokumentation zur Event-Publishing-Funktion für IMS und VSAM unter z/OS . . . . .	713
Dokumentation zur Event-Publishing-Funktion und Replikationsfunktion unter Linux, UNIX und Windows . . . . .	714
Dokumentation zur Funktion für Systeme zusammengeschnittener Datenbanken unter z/OS . . . . .	715
Dokumentation zur Funktion für Systeme zusammengeschnittener Datenbanken unter Linux, UNIX und Windows . . . . .	715
Dokumentation zu Enterprise Search unter Linux, UNIX und Windows . . . . .	716
Release-Informationen und Installationsvoraussetzungen . . . . .	717
<b>Eingabehilfen . . . . .</b>	<b>719</b>
Tastatureingabe und Navigation . . . . .	719
Tastatureingabe. . . . .	719
Navigation über die Tastatureingabe. . . . .	719
Tastatureingabebereich . . . . .	719
Eingabehilfen für Bildschirme . . . . .	720
Schriftarteneinstellungen . . . . .	720
Unabhängigkeit von Farben . . . . .	720
Kompatibilität mit Unterstützungseinrichtungen . . . . .	720
Dokumentation im behindertengerechten Format . . . . .	720
<b>Bemerkungen . . . . .</b>	<b>721</b>
Marken . . . . .	723
<b>Index . . . . .</b>	<b>725</b>
<b>Kontaktaufnahme mit IBM . . . . .</b>	<b>735</b>
Produktinformationen . . . . .	735
Kommentare zur Dokumentation. . . . .	735



---

## Zu diesem Handbuch

In diesem Handbuch wird erläutert, wie ein System zusammengeschlossener Datenbanken für den Zugriff auf Datenquellen konfiguriert wird. Dieses Handbuch enthält folgende Informationen:

- Eine Einführung zu den Konzepten, Komponenten und Funktionen von Systemen zusammengeschlossener Datenbanken.
- Empfehlungen zur Planung der Konfiguration der Datenquellen.
- Anweisungen für das Registrieren der Objekte, die erforderlich sind, damit der Server mit zusammengeschlossenen Datenbanken und die zusammengeschlossene Datenbank auf Datenquellen zugreifen können.
- Ausführliche Referenzinformationen zu den einzelnen Datenquellen: Informationen zur Serverkonfiguration, zu den Anforderungen für Umgebungsvariablen, zu den Datentypzuordnungen und zu den Optionen, die Sie definieren können, um die Konfiguration von Datenquellen anzupassen und zu optimieren.

### **Kennzeichnung zum Auffinden technischer Änderungen**

Technische Änderungen des Textes sind durch einen senkrechten Strich am linken Rand gekennzeichnet.

---

## Zielgruppe

Dieses Handbuch richtet sich an Systemadministratoren, Datenbankadministratoren, Sicherheitsadministratoren und Systembediener, die ein System zusammengeschlossener Datenbanken installieren und konfigurieren müssen. Mit Hilfe dieses Handbuchs können Sie ein System zusammengeschlossener Datenbanken für den Zugriff auf Daten aus relationalen und nicht relationalen Datenquellen verwalten. Dieses Handbuch kann auch von Programmierern und anderen Benutzern verwendet werden, die verstehen müssen, wie ein System zusammengeschlossener Datenbanken konfiguriert und verwendet wird. In diesem Handbuch wird davon ausgegangen, dass Sie sich mit DB2 auskennen. Sie sollten mit der standardmäßigen Datenbankterminologie vertraut sein und über Erfahrung mit dem Entwerfen und Verwalten von Datenbanken verfügen. In diesem Handbuch wird außerdem davon ausgegangen, dass Sie mit Ihren eigenen Anwendungen ebenso vertraut sind wie mit den Datenquellen, auf die Sie zugreifen wollen.

---

## In diesem Handbuch verwendete Konventionen und Terminologie

### **Terminologie für Systeme zusammengeschlossener Datenbanken:**

Das Glossar in diesem Handbuch definiert die Begriffe, die im Rahmen der Erörterung von Systemen zusammengeschlossener Datenbanken verwendet werden.

### **DB2-Befehle:**

In diesem Handbuch wird davon ausgegangen, dass DB2-Befehle - sofern nicht anders angegeben - über den DB2-Befehlszeilenprozessor (CLP) oder die grafische Benutzerschnittstelle (GUI) der DB2-Befehlszentrale eingegeben werden. Bei Nennung eines DB2-Befehls im Text werden lediglich der Befehl und dessen Parameter aufgeführt. Der Befehl wird nicht von DB2 verarbeitet.

## DB2-Steuerzentrale:

Die Dokumentation gibt an, wann Tasks über die DB2-Steuerzentrale ausgeführt werden können, und erläutert die erforderlichen Schritte zum Ausführen dieser Tasks.

## Hervorhebungsconventionen:

In diesem Handbuch werden die folgenden Hervorhebungsconventionen verwendet:

### Fettdruck

Hebt Befehle und Steuerelemente der grafischen Benutzerschnittstelle (beispielsweise Namen von Feldern, Namen von Druckknöpfen und Menüauswahlmöglichkeiten) hervor. Hinweise, Einschränkungen, Voraussetzungen und Empfehlungen werden ebenfalls durch Fettdruck gekennzeichnet.

### Monospace-Schrift

Hebt den vom Benutzer einzugebenden Text sowie Dateinamen und Codebeispiele hervor. Darüber hinaus wird die Monospaceschrift auch für die Parameternamen von SQL-Anweisungen und DB2-Befehlen verwendet.

### Kursivdruck

Hebt die Parameterwerte von SQL-Anweisungen und DB2-Befehlen hervor, die der Benutzer durch einen entsprechenden Wert ersetzen muss. Für Parameterwerte in Beispielen zu SQL-Anweisungen und DB2-Befehlen wird ebenfalls Kursivdruck verwendet. Darüber hinaus wird Kursivdruck verwendet, um bestimmte Wörter hervorzuheben, neue Begriffe zu identifizieren und Titel von Dokumenten anzugeben.

### GROSSBUCHSTABEN

Hebt die Namen von DB2-Befehlen und SQL-Anweisungen und deren Schlüsselwörter hervor. Die Namen von Datentypen, Optionen und Akronyme werden ebenfalls in Großbuchstaben angegeben.


---

## Lesen der Syntaxdiagramme

In diesem Handbuch wird die Syntax von Befehlen mit Hilfe einer Struktur beschrieben, die wie folgt definiert ist:

Lesen Sie die Syntaxdiagramme von links nach rechts und von oben nach unten, indem Sie dem Pfad der Linie folgen.

Das Symbol  gibt den Anfang einer Anweisung an.

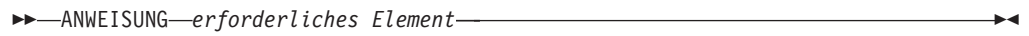
Das Symbol  gibt an, dass die Anweisungssyntax auf der nächsten Zeile fortgeführt wird.

Das Symbol  gibt an, dass eine Anweisung von der vorigen Zeile fortgeführt wird.

Das Symbol  gibt das Ende einer Anweisung an.

Erforderliche Elemente werden auf der horizontalen Linie (dem Hauptpfad) angezeigt.





Optionale Elemente werden unterhalb des Hauptpfads angezeigt.

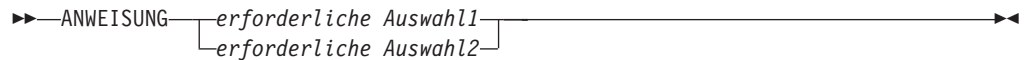


Wenn ein optionales Element oberhalb des Hauptpfads angegeben ist, hat dieses Element keine Auswirkung auf die Ausführung der Anweisung und wird nur aus Gründen der Lesbarkeit verwendet.

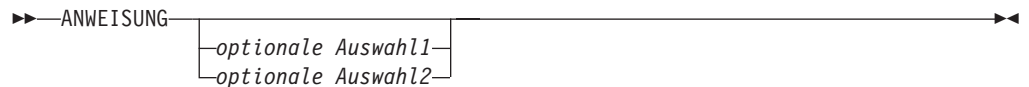


Wenn eine Auswahl zwischen zwei oder mehr Elementen möglich ist, werden die Elemente untereinander angezeigt.

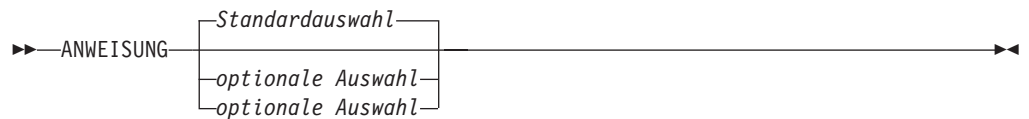
Wenn eines der Elemente ausgewählt werden *muss*, wird ein Element auf der Linie des Hauptpfads angezeigt.



Wenn es möglich ist, keines der Elemente auszuwählen, werden alle Elemente unterhalb des Hauptpfads angezeigt.



Wenn eines der Elemente der Standardwert ist, wird das Element oberhalb des Hauptpfads angegeben, während die übrigen Auswahlmöglichkeiten unterhalb des Hauptpfads angegeben werden.



Ein Pfeil über der Hauptlinie, der nach links zurückweist, gibt ein Element an, das wiederholt werden kann. In diesem Fall müssen wiederholte Elemente durch ein Leerzeichen oder mehrere Leerzeichen getrennt werden.



Wenn der Wiederholpfeil ein Komma enthält, müssen die wiederholten Elemente durch ein Komma getrennt werden.



Ein Wiederholpfeil über einem Elementstapel weist darauf hin, dass aus den untereinander angegebenen Elementen mehr als eine Auswahl angegeben oder eine einzige Auswahl mehrfach wiederholt werden kann.

Schlüsselwörter werden in Großbuchstaben (zum Beispiel FROM) angegeben. Sie müssen exakt wie gezeigt eingegeben werden. Variablen werden in Kleinbuchstaben (zum Beispiel spaltenname) angegeben. Sie stellen Namen oder Werte innerhalb der Syntax dar, die vom Benutzer anzugeben sind.

Wenn Satzzeichen, Klammern oder arithmetische Operatoren oder ähnliche Symbole im Syntaxdiagramm auftreten, müssen diese als Teil der Syntax eingegeben werden.

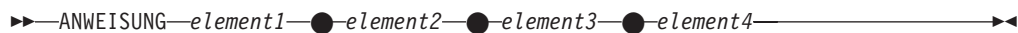
In einigen Fällen stellt ein einzige Variable eine Gruppe aus mehreren Parametern dar. Zum Beispiel kann im folgenden Diagramm in die Variable `parameterblock` jede der Interpretationen des Diagramms eingesetzt werden, die unter der Überschrift **parameterblock** angegeben werden:



**parameterblock:**



Benachbarte Segmente, die zwischen „großen Listenpunkten“ (●) angegeben werden, können in beliebiger Reihenfolge eingegeben werden.



Das obige Diagramm zeigt, dass `element2` und `element3` in beliebiger Reihenfolge eingegeben werden können. Beide der folgenden Eingaben sind demnach gültig:

```
ANWEISUNG element1 element2 element3 element4
ANWEISUNG element1 element3 element2 element4
```

---

## **Teil 1. Konzepte und Planung**



---

# Kapitel 1. Überblick über ein System zusammengesetzter Datenbanken

Die folgenden Abschnitte bieten eine Übersicht über ein System zusammengesetzter Datenbanken.

---

## System zusammengesetzter Datenbanken

Bei einem DB2<sup>®</sup>-*System zusammengesetzter Datenbanken* handelt es sich um einen speziellen Typ eines verteilten Datenbankverwaltungssystems (DBMS). Ein System zusammengesetzter Datenbanken besteht aus einem DB2-Exemplar, das als Server mit zusammengesetzten Datenbanken verwendet wird, einer Datenbank, die die zusammengesetzte Datenbank darstellt, einer oder mehrerer Datenquellen sowie Clients (Benutzer und Anwendungen), die auf die Datenbanken und Datenquellen zugreifen. Ein System zusammengesetzter Datenbanken ermöglicht die Versendung verteilter Anforderungen an mehrere Datenquellen mit einer einzigen SQL-Anweisung. Sie können beispielsweise Daten aus einer DB2 Universal Database<sup>™</sup>-Tabelle, einer Oracle-Tabelle und einer XML-Datei in einer einzigen SQL-Anweisung verknüpfen. In der folgenden Abbildung werden die Komponenten eines Systems zusammengesetzter Datenbanken und Beispiele für die Datenquellen gezeigt, auf die Sie zugreifen können.

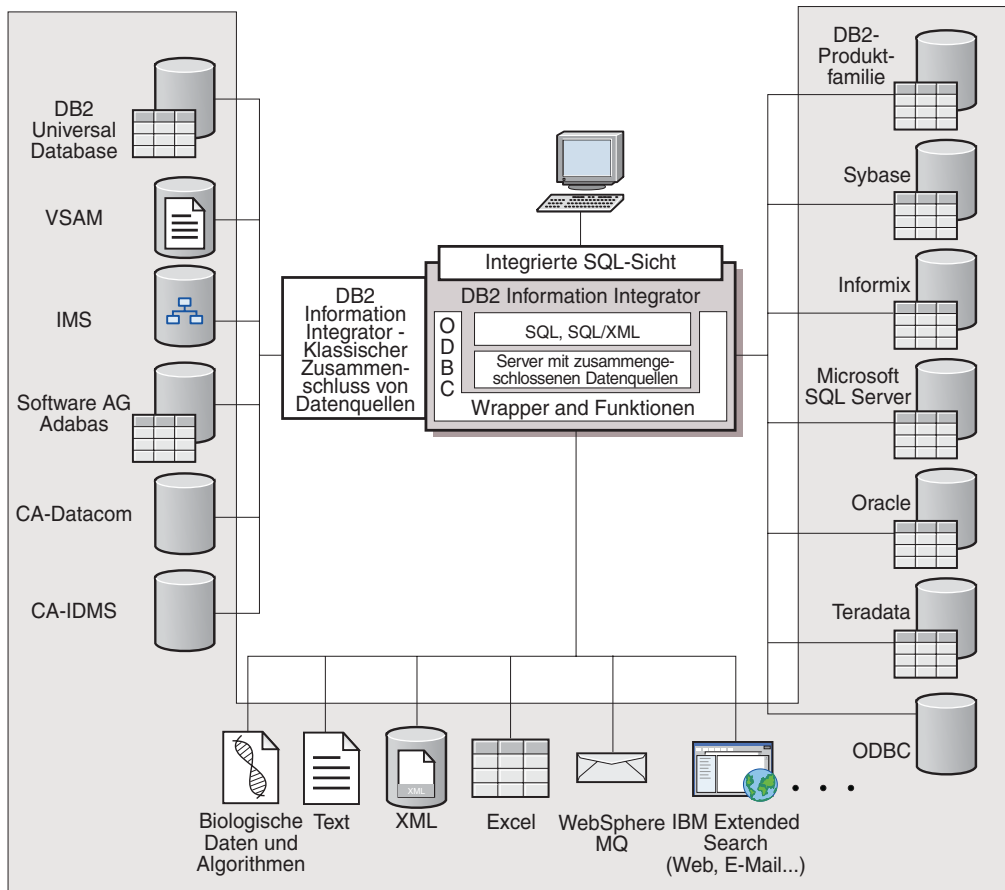


Abbildung 1. Komponenten eines Systems zusammenschlossener Datenbanken

Die Leistungsfähigkeit und Stärke eines DB2-Systems zusammenschlossener Datenbanken zeigt sich in folgenden Funktionen:

- Daten aus lokalen Tabellen und fernen Datenquellen können so miteinander verknüpft werden, als wären alle Daten lokal in der zusammenschlossenen Datenbank gespeichert.
- Daten in relationalen Datenquellen können aktualisiert werden, als wären die Daten in der zusammenschlossenen Datenbank gespeichert.
- Daten können in und aus relationalen Datenquellen repliziert werden.
- Die Verarbeitungsleistung der Datenquelle kann genutzt werden, indem Anforderungen zur Verarbeitung an die Datenquellen gesendet werden.
- SQL-Begrenzungen an der Datenquelle können durch die teilweise Verarbeitung einer verteilten Anforderung auf dem Server mit zusammenschlossenen Datenbanken kompensiert werden.

## Der Server mit zusammenschlossenen Datenbanken

Der DB2®-Server in einem System zusammenschlossener Datenbanken wird als *Server mit zusammenschlossenen Datenbanken* bezeichnet. Es kann eine beliebige Anzahl von DB2-Exemplaren als Server mit zusammenschlossenen Datenbanken konfiguriert werden. Sie können vorhandene DB2-Exemplare als Server mit zusammenschlossenen Datenbanken verwenden oder neue DB2-Exemplare speziell für das System zusammenschlossener Datenbanken erstellen.

Das DB2-Exemplar, das das System zusammengeschlossener Datenbanken verwaltet, wird als *Server* bezeichnet, weil es auf Anforderungen von Endbenutzern und Clientanwendungen antwortet. Der Server mit zusammengeschlossenen Datenbanken sendet häufig Teile der empfangenen Anforderungen zur Verarbeitung an die Datenquellen. Eine *Pushdown-Operation* ist eine Operation, die fern verarbeitet wird. Das DB2-Exemplar, das das System zusammengeschlossener Datenbanken verwaltet, wird als *Server mit zusammengeschlossenen Datenbanken* bezeichnet, auch wenn es beim Weiterleiten von Anforderungen im Pushdown-Modus an die Datenquellen als Client agiert.

Der Server mit zusammengeschlossenen Datenbanken ist wie jeder andere Anwendungsserver ein Datenbankmanagerexemplar. Anwendungsprozesse stellen eine Verbindung zur Datenbank innerhalb des Servers mit zusammengeschlossenen Datenbanken her und übergeben Anforderungen an die Datenbank. Der Server unterscheidet sich jedoch in zwei Hauptmerkmalen von anderen Anwendungsservern:

- Ein Server mit zusammengeschlossenen Datenbanken ist für den Empfang von Anforderungen konfiguriert, die ganz oder teilweise für Datenquellen bestimmt sein können. Der Server mit zusammengeschlossenen Datenbanken verteilt diese Anforderungen an die Datenquellen.
- Ein Server mit zusammengeschlossenen Datenbanken verwendet ähnlich wie andere Anwendungsserver DRDA<sup>®</sup>-Übertragungsprotokolle (über TCP/IP), um mit Exemplaren der DB2-Produktfamilie zu kommunizieren. Im Gegensatz zu anderen Anwendungsservern verwendet ein Server mit zusammengeschlossenen Datenbanken jedoch für den Zugriff auf die Datenquelle den nativen Client der Datenquelle. Ein Server mit zusammengeschlossenen Datenbanken verwendet beispielsweise für den Zugriff auf Sybase-Datenquellen Sybase Open Client und für den Zugriff auf Microsoft SQL Server-Datenquellen einen ODBC-Treiber für Microsoft<sup>®</sup> SQL Server.

#### **Zugehörige Konzepte:**

- „Was ist eine Datenquelle?“ auf Seite 5

---

## **Was ist eine Datenquelle?**

In einem System zusammengeschlossener Datenbanken kann eine *Datenquelle* ein relationales DBMS-Exemplar (z. B. Oracle oder Sybase) oder eine nicht relationale Datenquelle (z. B. ein BLAST-Suchalgorithmus oder eine XML-Datei) sein. Über einige Datenquellen ist auch der Zugriff auf weitere Datenquellen möglich. Beispielsweise kann über die Extended Search-Datenquelle auf Datenquellen wie Lotus<sup>®</sup> Notes-Datenbanken, Microsoft<sup>®</sup> Access, Microsoft Index Server, Internet-suchmaschinen und LDAP-Verzeichnisse (LDAP = Lightweight Directory Access Protocol) zugegriffen werden.

Die für den Zugriff auf eine Datenquelle verwendete Methode bzw. das hierzu verwendete Protokoll ist vom Datenquellentyp abhängig. Beispielsweise wird DRDA<sup>®</sup> für den Zugriff auf Datenquellen von DB2<sup>®</sup> für z/OS<sup>™</sup> und OS/390<sup>®</sup> und die Documentum-Client-API/-Bibliothek für den Zugriff auf Documentum-Datenquellen verwendet.

Datenquellen sind halbautonom. Der Server mit zusammengeschlossenen Datenbanken kann z. B. Anfragen an Oracle-Datenquellen senden, während diese Oracle-Anwendungen auf diese Datenquellen zugreifen. Ein DB2-System zusammengeschlossener Datenbanken monopolisiert oder begrenzt den Zugriff auf andere Datenquellen außerhalb der Integritäts- und Verriegelungsbedingungen nicht.

**Zugehörige Konzepte:**

- „Die zusammengeschlossene Datenbank“ auf Seite 8

**Zugehörige Referenzen:**

- „Unterstützte Datenquellen“ auf Seite 6

---

## Unterstützte Datenquellen

Es gibt viele Datenquellen, auf die Sie über ein System zusammengeschlossener Datenbanken zugreifen können. In der folgenden Tabelle sind die unterstützten Datenquellen aufgeführt.

*Tabelle 1. Unterstützte Datenquellenversionen und Zugriffsmethoden.*

Datenquelle	Unterstützte Versionen	Zugriffsmethode
DB2 Universal Database™ für Linux, UNIX und Windows®	7.2, 8.1, 8.2	DRDA®
DB2 Universal Database für z/OS™ und OS/390®	6.1, 7.1 mit den folgenden angewendeten APARs: <ul style="list-style-type: none"> <li>• PQ62695</li> <li>• PQ55393</li> <li>• PQ56616</li> <li>• PQ54605</li> <li>• PQ46183</li> <li>• PQ62139</li> </ul> 8.1	DRDA
DB2 Universal Database für iSeries™	5.1 <ul style="list-style-type: none"> <li>• mit den folgenden angewendeten APARs: <ul style="list-style-type: none"> <li>– SE06003</li> <li>– SE06872</li> <li>– II13348</li> </ul> </li> <li>• mit den folgenden angewendeten PTFs: <ul style="list-style-type: none"> <li>– SI05990</li> <li>SI05991</li> </ul> </li> </ul> 5.2 mit angewendeter PTF SI0735.	DRDA
DB2 Server für VM und VSE	7.1 (oder höher) mit angewendeten Korrekturen für APARs für Schemafunktionen.	DRDA
Informix™	7.31, 8.32, 8.4, 9.3, 9.4	Informix Client SDK V2.7 (oder höher)
ODBC	3.x	ODBC-Treiber für die Datenquelle, z. B. ODBC-Treiber von Redbrick für den Zugriff auf Redbrick
OLE DB	2.7, 2.8	OLE DB 2.0 (oder höher)



Tabelle 1. Unterstützte Datenquellenversionen und Zugriffsmethoden. (Forts.)

Datenquelle	Unterstützte Versionen	Zugriffsmethode
Oracle	8.0.6, 8.1.6, 8.1.7, 9.0, 9.1, 9.2, 9i, 10g	Oracle-Netzclient oder NET8-Client-Software
Microsoft SQL Server	7.0, 2000 SP3 und neuere Service-Packs für dieses Release	Unter Windows, Microsoft SQL Server-Client-Treiber für ODBC 3.0 (oder höher)  Unter UNIX, DataDirect Technologies (früher MERANT) Connect-Treiber für ODBC 3.7 (oder höher)
Sybase	11.9.2, 12.x	CTLIB-Schnittstelle für Sybase Open Client
Teradata	V2R3, V2R4, V2R5	Teradata Call-Level Interface, Version 2 (CLIV2) Release 04.06 (oder höher)
BLAST	2.2.3 und spätere Fix-packs für Version 2.2 werden unterstützt	BLAST-Dämon (wird mit dem Wrapper zur Verfügung gestellt)
BioRS	v5.0.14	keine
Documentum	3.x, 4.x	Documentum Client library/APL3.1.7a (oder höher)
Entrez (PubMed- und GenBank-Datenquellen)	1.0	keine
HMMER	2.2g, 2.3	HMMER-Dämon (wird mit dem Wrapper zur Verfügung gestellt)
IBM Lotus Extended Search	4.0.1, 4.0.2	Extended Search-Clientbibliothek (wird mit dem Wrapper zur Verfügung gestellt)
Microsoft Excel	97, 2000, 2002, 2003	Excel 97, 2000, 2002 oder 2003 auf dem Server mit zusammengeschlossenen Datenbanken installiert
PeopleSoft	8.x	IBM WebSphere Business Integration-Adapter für PeopleSoft v2.3.1, 2.4
SAP	3.x, 4.x	IBM WebSphere Business Integration-Adapter für mySAP.com v2.3.1, 2.4
Siebel	7, 7.5, 2000	IBM WebSphere Business Integration-Adapter für Siebel eBusiness Applications v2.3.1, 2.4
Dateien mit Tabellenstruktur		keine
Benutzerdefinierte Funktionen für KEGG	Unterstützt	

Tabelle 1. Unterstützte Datenquellenversionen und Zugriffsmethoden. (Forts.)

Datenquelle	Unterstützte Versionen	Zugriffsmethode
Benutzerdefinierte Funktionen für Life Sciences	Unterstützt	
Web-Services	Spezifikationen für SOAP 1.0., 1.1, WSDL 1.0, 1.1	HTTP
XML	1.0-Spezifikation	keine

**Zugehörige Konzepte:**

- „Was ist eine Datenquelle?“ auf Seite 5

---

## Die zusammengeschlossene Datenbank

Für Endbenutzer und Clientanwendungen werden die Datenquellen in DB2® als eine einzige, kollektive Datenbank dargestellt. Benutzer und Anwendungen verfügen über eine Schnittstelle zur *zusammengeschlossenen Datenbank*, die vom Server mit zusammengeschlossenen Datenbanken verwaltet wird. Die zusammengeschlossene Datenbank enthält einen Systemkatalog. Der Katalog des Systems zusammengeschlossener Datenbanken enthält Einträge, in denen die Datenquellen und deren Kenndaten angegeben sind. Der Server mit zusammengeschlossenen Datenbanken ermittelt anhand der im Systemkatalog der zusammengeschlossenen Datenbank gespeicherten Informationen die optimale Methode für die Verarbeitung von SQL-Anweisungen.

Das System zusammengeschlossener Datenbanken verarbeitet SQL-Anweisungen so, als ob die Datenquellen gewöhnliche relationale Tabellen oder Sichten innerhalb der zusammengeschlossenen Datenbank wären. Dadurch ist Folgendes möglich:

- Das System zusammengeschlossener Datenbanken kann relationale und nicht relationale Daten verknüpfen. Dies gilt sogar dann, wenn Datenquellen unterschiedliche SQL-Programmversionen verwenden oder SQL überhaupt nicht unterstützen.
- Die Kenndaten der zusammengeschlossenen Datenbank haben Vorrang, wenn zwischen den Kenndaten der zusammengeschlossenen Datenbank und den Kenndaten der Datenquellen Unterschiede bestehen:
  - Angenommen, der Server mit zusammengeschlossenen Datenbanken verwendet eine andere Codepage als die Datenquelle. Wenn an einen Benutzer des Servers mit zusammengeschlossenen Datenbanken Zeichendaten von der Datenquelle zurückgegeben werden, werden diese Daten auf Basis der Codepage umgewandelt, die die zusammengeschlossene Datenbank verwendet.
  - Angenommen, der Server mit zusammengeschlossenen Datenbanken verwendet eine andere Sortierfolge als die Datenquelle. Alle Sortieroperationen für Zeichendaten werden auf dem Server mit zusammengeschlossenen Datenbanken und nicht in der Datenquelle ausgeführt.

**Zugehörige Konzepte:**

- „Der SQL-Compiler“ auf Seite 10
- „Der Systemkatalog der zusammengeschlossenen Datenbank“ auf Seite 9

---

## Der Systemkatalog der zusammengeschlossenen Datenbank

Der Systemkatalog der zusammengeschlossenen Datenbank enthält Informationen über Objekte in der zusammengeschlossenen Datenbank und über Objekte in den Datenquellen. Der Katalog in einer zusammengeschlossenen Datenbank wird als *globaler Katalog* bezeichnet, da er Informationen über das gesamte System zusammengeschlossener Datenbanken enthält. Das DB2<sup>®</sup>-Abfrageoptimierungsprogramm verwendet die Informationen des globalen Katalogs und des Datenquellenwrappers, um die beste Möglichkeit für die Verarbeitung von SQL-Anweisungen zu ermitteln. Die im globalen Katalog gespeicherten Informationen beinhalten ferne und lokale Informationen, z. B. Spaltennamen, Spaltentypen, Spaltenstandardwerte und Indexinformationen.

*Ferne* Kataloginformationen sind die Informationen oder Namen, die von der Datenquelle verwendet werden. *Lokale* Kataloginformationen sind die Informationen oder Namen, die von der zusammengeschlossenen Datenbank verwendet werden. Beispiel: Eine ferne Tabelle enthält eine Spalte mit dem Namen *EMPNO*. Der globale Katalog speichert den fernen Spaltennamen als *EMPNO*. Wenn Sie keinen anderen Namen vergeben, wird der lokale Spaltenname als *EMPNO* gespeichert. Sie können den lokalen Spaltennamen in *Employee\_Number* ändern. Wenn Benutzer Abfragen übergeben, die diese Spalte enthalten, dann müssen sie in den Abfragen an Stelle von *EMPNO* die Spalte *Employee\_Number* verwenden. Verwenden Sie die Anweisung *ALTER NICKNAME*, um den lokalen Namen der Datenquellenspalten zu ändern.

Bei relationalen Datenquellen enthalten die im globalen Katalog gespeicherten Informationen sowohl ferne als auch lokale Informationen.

Bei nicht relationalen Datenquellen sind die im globalen Katalog gespeicherten Informationen von der Datenquelle abhängig.

Zum Anzeigen der im globalen Katalog gespeicherten Tabelleninformationen der Datenquelle fragen Sie die Katalogsichten *SYSCAT.TABLES*, *SYSCAT.TABOPTIONS*, *SYSCAT.INDEXES*, *SYSCAT.COLUMNS* und *SYSCAT.COLOPTIONS* in der zusammengeschlossenen Datenbank ab.

Im globalen Katalog sind auch andere Informationen über die Datenquellen enthalten. Dies sind beispielsweise Informationen, die der Server mit zusammengeschlossenen Datenbanken für die Verbindung zur Datenquelle und für die Zuordnung zwischen Benutzerberechtigungen für die zusammengeschlossene Datenbank und Benutzerberechtigungen für die Datenquelle verwendet. Der globale Katalog enthält Attribute der Datenquelle, die Sie explizit festgelegt haben, beispielsweise Serveroptionen.

### Zugehörige Konzepte:

- „Der SQL-Compiler“ auf Seite 10

### Zugehörige Referenzen:

- Anhang A, „Sichten in der globalen Katalogtabelle mit Informationen der zusammengeschlossenen Datenbank“, auf Seite 629

---

## Der SQL-Compiler

Um Daten von Datenquellen abzurufen, übergeben Benutzer und Anwendungen Abfragen in DB2<sup>®</sup>-SQL an die zusammengeschlossene Datenbank. Wenn die Abfrage übergeben wird, ruft der DB2-SQL-Compiler die Informationen im globalen Katalog und im Wrapper der Datenquelle ab, um die Verarbeitung der Abfrage zu unterstützen. Dabei handelt es sich um Informationen über die Verbindung zur Datenquelle, Serverattribute, Zuordnungen, Indexinformationen und Verarbeitungsstatistiken.

### Zugehörige Konzepte:

- „Wrapper und Wrappermodule“ auf Seite 13
- „Das Abfrageoptimierungsprogramm“ auf Seite 10

---

## Das Abfrageoptimierungsprogramm

Das *Abfrageoptimierungsprogramm* analysiert im Rahmen des SQL-Compilerprozesses die Abfrage. Der Compiler entwickelt alternative Strategien für die Verarbeitung der Abfrage. Diese Strategien werden als *Zugriffspläne* bezeichnet. Die Zugriffspläne können die Abfrage unter folgenden Bedingungen aufrufen:

- Die Verarbeitung der Abfrage soll durch die Datenquellen erfolgen.
- Die Verarbeitung der Abfrage soll durch den Server mit zusammengeschlossenen Datenbanken erfolgen.
- Die Verarbeitung der Abfrage soll teilweise durch die Datenquellen und teilweise durch den Server mit zusammengeschlossenen Datenbanken erfolgen.

DB2<sup>®</sup> UDB wertet die Zugriffspläne in erster Linie auf der Grundlage von Informationen über das Leistungsspektrum der Datenquelle und der Daten aus. Diese Informationen sind im Wrapper und im globalen Katalog enthalten. DB2 UDB zerlegt die Abfrage in Segmente, die als *Abfragefragmente* bezeichnet werden. Normalerweise ist es effizienter, ein Abfragefragment an eine Datenquelle *im Pushdown-Modus weiterzuleiten*, wenn die Datenquelle das Fragment verarbeiten kann. Das Abfrageoptimierungsprogramm berücksichtigt jedoch andere Faktoren wie z. B. die im Folgenden aufgeführten:

- Die zu verarbeitende Datenmenge
- Die Verarbeitungsgeschwindigkeit der Datenquelle
- Die vom Fragment zurückgegebene Datenmenge
- Die Bandbreite der Übertragung
- Das Vorhandensein einer verwertbaren gespeicherten Abfragetabelle auf dem Server mit zusammengeschlossenen Datenbanken, die die gleichen Abfrageergebnisse darstellt

Das Abfrageoptimierungsprogramm generiert auf der Grundlage des Ressourcenaufwands lokale und ferne Zugriffspläne für die Verarbeitung eines Abfragefragments. DB2 UDB wählt dann den Plan mit dem wahrscheinlich niedrigsten Ressourcenaufwand für die Verarbeitung der Abfrage aus.

Sollen Fragmente von Datenquellen verarbeitet werden, übergibt DB2 UDB diese Fragmente an die Datenquellen. Nachdem die Datenquellen die Fragmente verarbeitet haben, werden die Ergebnisse abgerufen und an DB2 UDB zurückgegeben. Hat DB2 UDB einen Teil der Verarbeitung durchgeführt, kombiniert DB2 UDB

seine Ergebnisse mit den von der Datenquelle abgerufenen Ergebnissen. DB2 UDB gibt dann alle Ergebnisse an den Client zurück.

**Zugehörige Konzepte:**

- „Der SQL-Compiler“ auf Seite 10
- „Kompensation“ auf Seite 11
- „Optimieren der Abfrageverarbeitung“ in *Systeme zusammenschlossener Datenbanken*

---

## Kompensation

Der DB2®-Server mit zusammenschlossenen Datenbanken leitet ein Abfragefragment nicht im Pushdown-Modus weiter, wenn die Datenquelle es nicht verarbeiten kann oder wenn der Server mit zusammenschlossenen Datenbanken es schneller verarbeiten kann als die Datenquelle. Beispiel: Die SQL-Programmversion der Datenquelle unterstützt keine Gruppierung CUBE in der Klausel GROUP BY. Enthält eine Abfrage eine Gruppierung CUBE und verweist diese Abfrage auf eine Tabelle in dieser Datenquelle, wird diese Abfrage an den Server mit zusammenschlossenen Datenbanken übergeben. DB2 Information Integrator leitet die Gruppierung CUBE nicht im Pushdown-Modus an die Datenquelle weiter, sondern führt die CUBE-Verarbeitung selbst durch. Die Fähigkeit von DB2 Information Integrator, von einer Datenquelle nicht unterstützten SQL-Code zu verarbeiten, wird als *Kompensation* bezeichnet.

Der Server mit zusammenschlossenen Datenbanken kompensiert die nicht ausreichende Funktionalität der Datenquelle auf zwei Arten:

- Er fordert die Datenquelle auf, eine oder mehrere Verarbeitungsoperationen zu verwenden, die der DB2-Funktion in der Abfrage entsprechen. Angenommen, eine Datenquelle unterstützt nicht die Kotangensfunktion (COT(x)), sie unterstützt jedoch die Tangensfunktion (TAN(x)). DB2 Information Integrator kann die Datenquelle auffordern, die Berechnung durchzuführen (1/TAN(x)), die der Kotangensfunktion (COT(x)) entspricht.
- Die Datensätze können an den Server mit zusammenschlossenen Datenbanken zurückgegeben werden, und die Funktion wird lokal ausgeführt.

| Bei relationalen Datenquellen unterstützen alle Typen von Verwaltungssystemen  
| für relationale Datenbanken (RDBMS) eine Untergruppe des internationalen SQL-  
| Standards. Manche RDBMS-Typen unterstützen zusätzlich SQL-Konstrukte, die  
| über diesen Standard hinausgehen. Eine *SQL-Programmversion* ist die Gesamtheit  
| des SQL, das ein RDBMS-Typ unterstützt. Wird ein SQL-Konstrukt in der DB2-  
| SQL-Programmversion gefunden, nicht jedoch in der Programmversion der relationalen  
| Datenquelle, kann der Server mit zusammenschlossenen Datenbanken dieses  
| Konstrukt für die Datenquelle implementieren.

DB2 Information Integrator kann Unterschiede zwischen SQL-Programmversionen kompensieren. Ein Beispiel für diese Fähigkeit ist die Klausel 'allgemeiner Tabellenausdruck'. DB2-SQL schließt die Klausel 'allgemeiner Tabellenausdruck' mit ein. In dieser Klausel kann ein Name angegeben werden, mit dem alle Klauseln FROM in einer Gesamtauswahl auf eine Ergebnismenge verweisen können. Der Server mit zusammenschlossenen Datenbanken verarbeitet dann einen allgemeinen Tabellenausdruck für eine Datenquelle, obwohl die SQL-Programmversion der Datenquelle allgemeine Tabellenausdrücke nicht mit einschließt.

Durch die Kompensation kann der Server mit zusammengeschlossenen Datenbanken bei Abfragen von Datenquellen die vollständige DB2-SQL-Programmversion unterstützen. Die Vorteile dieser Kompensation gelten auch für Datenquellen, die SQL nur eingeschränkt oder überhaupt nicht unterstützen. In einem System zusammengeschlossener Datenbanken muss - außer bei einer Durchgriffssitzung - die DB2-SQL-Programmversion verwendet werden.

**Zugehörige Konzepte:**

- „Durchgriffssitzungen“ auf Seite 12

---

## Durchgriffssitzungen

Sie können SQL-Anweisungen direkt an Datenquellen übergeben, indem Sie einen speziellen Modus verwenden, der *Durchgriff* genannt wird. Sie übergeben SQL-Anweisungen in der SQL-Programmversion, die von der Datenquelle verwendet wird. Verwenden Sie eine Durchgriffssitzung, wenn Sie eine Verarbeitung ausführen möchten, die von der DB2<sup>®</sup>-SQL oder -API nicht unterstützt wird. Verwenden Sie beispielsweise eine Durchgriffssitzung, um eine Prozedur oder einen Index zu erstellen oder um Abfragen in der Programmversion der Datenquelle durchzuführen.

Momentan unterstützen die Datenquellen mit Unterstützung für die Durchgriffsfunktion diese Funktion über SQL. Zukünftig wird es möglich sein, dass Datenquellen Durchgriffssitzungen auch mit anderen Sprachen unterstützen.

Sie können auch eine Durchgriffssitzung verwenden, um nicht durch SQL unterstützte Aktionen auszuführen, z. B. bestimmte administrative Tasks. Sie können jedoch nicht alle administrativen Tasks mit Hilfe einer Durchgriffssitzung durchführen. Sie können beispielsweise Tabellen in der Datenquelle erstellen oder löschen. Sie können jedoch die ferne Datenbank nicht starten oder stoppen.

Sie können in einer Durchgriffssitzung sowohl statisches als auch dynamisches SQL verwenden.

Der Server mit zusammengeschlossenen Datenbanken stellt die folgenden SQL-Anweisungen für die Verwaltung von Durchgriffssitzungen zur Verfügung:

**SET PASSTHRU**

Eröffnet eine Durchgriffssitzung. Geben Sie eine weitere Anweisung SET PASSTHRU aus, um eine neue Durchgriffssitzung zu starten, wird die aktuelle Durchgriffssitzung beendet.

**SET PASSTHRU RESET**

Beendet die aktuelle Durchgriffssitzung.

**GRANT (Server Privileges)**

Erteilt einem Benutzer, einer Gruppe, einer Liste von Berechtigungs-IDs oder PUBLIC die Berechtigung, Durchgriffssitzungen zu einer bestimmten Datenquelle einzuleiten.

**REVOKE (Server Privileges)**

Widerruft die Berechtigung zur Einleitung von Durchgriffssitzungen.



Für Durchgriffssitzungen gelten folgende Beschränkungen:

- Sie müssen die SQL-Programmversion oder die Sprachbefehle der Datenquelle verwenden - Sie können nicht die DB2-SQL-Programmversion verwenden. Daher fragen Sie nicht einen Kurznamen ab, sondern direkt die Objekte der Datenquelle.
- Beim Durchführen der Verarbeitungsoperationen für UPDATE oder DELETE in einer Durchgriffssitzung können Sie nicht die Bedingung WHERE CURRENT OF CURSOR verwenden.
- LOBs werden in Durchgriffssitzungen nicht unterstützt.

#### Zugehörige Konzepte:

- „Wrapper und Wrappermodule“ auf Seite 13
- „Direktes Abfragen von Datenquellen im Durchgriffsmodus“ in *Systeme zusammengeschnittener Datenbanken*

---

## Wrapper und Wrappermodule

*Wrapper* umfassen Mechanismen, mit deren Hilfe Server mit zusammengeschnittenen Datenbanken mit den verwendeten Datenquellen interagieren können. Der Server mit zusammengeschnittenen Datenbanken verwendet für die Implementierung eines Wrappers Routinen, die in einer als *Wrappermodul* bezeichneten Bibliothek gespeichert sind. Diese Routinen ermöglichen dem Server die Ausführung von Operationen wie das Herstellen einer Verbindung zu einer Datenquelle und das interaktive Abrufen von Daten von dieser Datenquelle. Der DB2<sup>®</sup>-Exemplareigner verwendet normalerweise die Anweisung CREATE WRAPPER, um einen Wrapper im System zusammengeschnittener Datenbanken zu registrieren. Sie können einen Wrapper mit der Wrapperoption DB2\_FENCED als abgesichert oder akzeptiert registrieren.

Sie erstellen einen Wrapper für jeden Datenquellentyp, auf den Sie zugreifen möchten. Beispiel: Sie möchten auf drei Datenbanktabellen von DB2 für z/OS<sup>™</sup>, eine Tabelle von DB2 für iSeries<sup>™</sup>, zwei Tabellen von Informix<sup>®</sup> und eine Sicht von Informix zugreifen. Sie müssen einen Wrapper für DB2-Datenquellenobjekte und einen Wrapper für Informix-Datenquellenobjekte erstellen. Sobald diese Wrapper in der zusammengeschnittenen Datenbank registriert sind, können Sie diese für den Zugriff auf andere Objekte in diesen Datenquellen verwenden. Sie können beispielsweise den DRDA<sup>®</sup>-Wrapper für alle Datenquellenobjekte der DB2-Produktfamilie, d. h. für DB2 für Linux, UNIX<sup>®</sup> und Windows<sup>®</sup>, DB2 für z/OS und OS/390<sup>®</sup>, DB2 für iSeries und DB2 Server für VM und VSE verwenden.

Verwenden Sie die Serverdefinitionen und Kurznamen zur Angabe der technischen Daten (Name, Speicherposition, usw.) jedes Datenquellenobjekts.

Ein Wrapper dient zur Ausführung zahlreicher Tasks. Hierzu gehören z. B. die im Folgenden aufgeführten Operationen:

- Er stellt die Verbindung zur Datenquelle her. Der Wrapper verwendet die Standardverbindungs-API der Datenquelle.
- Er übergibt Abfragen an die Datenquelle.
  - Bei Datenquellen, die SQL unterstützen, wird die Abfrage in SQL übergeben.
  - Bei Datenquellen, die SQL nicht unterstützen, wird die Abfrage in die native Abfragesprache der Quelle oder in eine Reihe von Quellen-API-Aufrufen umgesetzt.

- Er empfängt Ergebnismengen von der Datenquelle. Der Wrapper verwendet die Standard-APIs der Datenquellen, um Ergebnismengen zu empfangen.
- Er antwortet auf Abfragen des Servers mit zusammengeschlossenen Datenbanken zu den Standarddatentypzuordnungen für eine Datenquelle. Der Wrapper enthält die Standardtypzuordnungen, die zur Erstellung von Kurznamen für ein Datenquellenobjekt verwendet werden. Bei relationalen Wrappern werden beim Erstellen von Datentypzuordnungen die Standarddatentypzuordnungen außer Kraft gesetzt. Benutzerdefinierte Datentypzuordnungen werden im globalen Katalog gespeichert.
- Er antwortet auf Abfragen des Servers mit zusammengeschlossenen Datenbanken zu den Standardfunktionszuordnungen für eine Datenquelle. Der Wrapper enthält Informationen, die der Server mit zusammengeschlossenen Datenbanken benötigt, um festzustellen, ob und wie DB2-Funktionen zu Funktionen der Datenquelle zugeordnet werden. Diese Informationen verwendet der SQL-Compiler, um festzustellen, ob die Datenquelle die Abfrageoperationen ausführen kann. Bei relationalen Wrappern werden beim Erstellen von Funktionszuordnungen die Standardfunktionszuordnungen außer Kraft gesetzt. Benutzerdefinierte Funktionszuordnungen werden im globalen Katalog gespeichert.

*Wrapperoptionen* werden zur Konfiguration des Wrappers oder zum Definieren der Art und Weise verwendet, in der DB2 Information Integrator den Wrapper einsetzt.

#### Zugehörige Tasks:

- „Trusted and fenced mode process environments“ in *IBM DB2 Information Integrator Wrapper Developer's Guide*
- „Registrieren von Wrappern für eine Datenquelle“ auf Seite 70

#### Zugehörige Referenzen:

- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*
- Anhang B, „Wrapperoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 633

## Standardnamen für Wrapper

Für jede unterstützte Datenquelle gibt es Wrapper. Einige Wrapper haben Standardnamen. Wenn Sie für die Erstellung des Wrappers den Standardnamen verwenden, wählt der Server mit zusammengeschlossenen Datenbanken automatisch die dem Wrapper zugeordnete Bibliothek der Datenquelle aus.

*Tabelle 2. Standardnamen der Wrapper für alle Datenquellen.*

Datenquelle	Standardnamen der Wrapper
DB2 Universal Database™ für Linux, UNIX und Windows®	DRDA
DB2 Universal Database für z/OS und OS/390®	DRDA
DB2 Universal Database für iSeries	DRDA
DB2 Server für VM und VSE	DRDA
Informix	INFORMIX
Microsoft® SQL Server	MSSQLODBC3
ODBC	ODBC
OLE DB	OLEDB



Tabelle 2. Standardnamen der Wrapper für alle Datenquellen. (Forts.)

Datenquelle	Standardnamen der Wrapper
Oracle	NET8
Sybase	CTLIB
Teradata	TERADATA
BLAST	keine
BioRS	keine
Documentum	keine
Entrez	keine
Extended Search	keine
HMMER	keine
Microsoft Excel	keine
Dateien mit Tabellenstruktur	keine
Web-Services	keine
WebSphere Business Integration	keine
XML	keine

**Zugehörige Konzepte:**

- „Wrapper und Wrappermodule“ auf Seite 13

---

## Serverdefinitionen und Serveroptionen

Nach der Erstellung des Wrappers für die Datenquelle gibt der Exempleigner die Datenquellen für die zusammengeschlossene Datenbank an. Der Exempleigner stellt einen Namen zur Identifizierung der Datenquelle sowie weitere die Datenquelle betreffende Informationen zur Verfügung. Diese Informationen umfassen Folgendes:

- Typ und Version der Datenquelle
- Datenbanknamen der Datenquelle (nur RDBMS)
- Spezifische Metadaten der Datenquelle

Eine Datenquelle der DB2<sup>®</sup>-Produktfamilie kann beispielsweise mehrere Datenbanken haben. In der Definition muss angegeben werden, zu welcher Datenbank der Server mit zusammengeschlossenen Datenbanken eine Verbindung herstellen kann. Im Gegensatz dazu hat eine Oracle-Datenquelle eine Datenbank. Der Server mit zusammengeschlossenen Datenbanken kann zu dieser Datenbank eine Verbindung herstellen, ohne ihren Namen zu kennen. Der Name der Datenbank ist in der Serverdefinition des Servers mit zusammengeschlossenen Datenbanken für eine Oracle-Datenquelle nicht mit eingeschlossen.

Der Name und die anderen Informationen, die der Exempleigner dem Server mit zusammengeschlossenen Datenbanken zur Verfügung stellt, werden zusammen als *Serverdefinition* bezeichnet. Datenquellen antworten auf Datenanforderungen und sind Server mit eigenen Rechten.

Die Anweisungen CREATE SERVER und ALTER SERVER werden verwendet, um eine Serverdefinition zu erstellen oder zu ändern.

Einige der in einer Serverdefinition enthaltenen Informationen sind als *Serveroptionen* gespeichert. Für das Erstellen von Serverdefinitionen ist das Verständnis der Optionen, die für den Server angegeben werden können, wichtig. Einige Serveroptionen dienen zum Konfigurieren des Wrappers, andere zur Steuerung der Art und Weise, in der der Wrapper von DB2 Information Integrator verwendet wird.

Serveroptionen können entweder für die Dauer einer einzelnen Verbindung oder für mehrere aufeinanderfolgende Verbindungen festgelegt werden.

**Zugehörige Tasks:**

- „Registrieren von Serverdefinitionen für eine Datenquelle“ auf Seite 71

**Zugehörige Referenzen:**

- „ALTER SERVER statement“ in *SQL Reference, Volume 2*
- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641

---

## Benutzerzuordnungen

Wenn ein Server mit zusammenschlossenen Datenbanken im Pushdown-Modus eine Abfrage an eine Datenquelle weiterleitet, muss er zunächst eine Verbindung zu dieser Datenquelle aufbauen.

Bei den meisten Datenquellen verwendet der Server mit zusammenschlossenen Datenbanken dafür eine für diese Datenquelle gültige Benutzer-ID und ein gültiges Kennwort. Wenn für die Verbindung zu einer Datenquelle eine Benutzer-ID und ein Kennwort erforderlich sind, können Sie eine Zuordnung zwischen der Berechtigungs-ID des Servers mit zusammenschlossenen Datenbanken und der Benutzer-ID der Datenquelle mit dem zugehörigen Kennwort definieren. Diese Zuordnung kann für alle Benutzer-IDs erstellt werden, die das System zusammenschlossener Datenbanken zum Senden verteilter Anforderungen benutzen. Diese Zuordnung wird als *Benutzerzuordnung* bezeichnet.

In bestimmten Fällen müssen Sie keine Benutzerzuordnung erstellen, wenn die Benutzer-ID und das Kennwort, die zur Herstellung der Verbindung zur zusammenschlossenen Datenbank verwendet werden, mit den entsprechenden Angaben übereinstimmen, die für den Zugriff auf die ferne Datenquelle verwendet werden.

**Zugehörige Tasks:**

- „Registrieren von Benutzerzuordnungen für eine Datenquelle“ auf Seite 72

**Zugehörige Referenzen:**

- „ALTER USER MAPPING statement“ in *SQL Reference, Volume 2*
- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- Anhang E, „Benutzerzuordnungsoptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 659

---

## Kurznamen und Datenquellenobjekte

Nachdem Sie die Serverdefinitionen und Benutzerzuordnungen erstellt haben, erstellt der Exemplareigner der zusammengeschlossenen Datenbank die Kurznamen. Ein *Kurzname* ist eine Kennung, die auf das Objekt in der Datenquelle verweist, auf das Sie zugreifen möchten. Die durch Kurznamen angegebenen Objekte werden als *Datenquellenobjekte* bezeichnet.

Bei Kurznamen handelt es sich nicht um alternative Namen für Datenquellenobjekte im Sinne von Aliasnamen. Sie sind Zeiger, mit denen der Server mit zusammengeschlossenen Datenbanken auf diese Objekte verweist. Kurznamen werden normalerweise mit der Anweisung CREATE NICKNAME und bestimmten Kurznamenspaltoptionen und Kurznamenoptionen definiert.

Übergibt ein Endbenutzer oder eine Clientanwendung eine verteilte Anforderung an den Server mit zusammengeschlossenen Datenbanken, muss die Anforderung die Datenquellen nicht angeben. Stattdessen verweist die Anforderung anhand der Kurznamen auf die Datenquellenobjekte. Kurznamen werden bestimmten Objekten in der Datenquelle zugeordnet. Durch diese Zuordnungen ist keine Qualifikation der Kurznamen durch Datenquellennamen mehr erforderlich. Die Speicherposition der Datenquellenobjekte ist für die Clientanwendung oder den Endbenutzer transparent.

Angenommen, Sie definieren den Kurznamen *DEPT*, der eine Informix®-Datenbanktabelle des Namens *NFX1.PERSON* darstellt. Der Server mit zusammengeschlossenen Datenbanken akzeptiert die Anweisung SELECT \* FROM *DEPT*. Die Verwendung der Anweisung SELECT \* FROM *NFX1.PERSON* über den Server mit zusammengeschlossenen Datenbanken ist (außer bei Durchgriffssitzungen) jedoch nicht zulässig, sofern auf diesem Server nicht eine lokale Tabelle mit dem Namen *NFX1.PERSON* vorhanden ist.

Beim Erstellen eines Kurznamens für ein Datenquellenobjekt werden dem globalen Katalog Metadaten über das Objekt hinzugefügt. Das Abfrageoptimierungsprogramm verwendet diese Metadaten und die Informationen im Wrapper, um den Zugriff auf das Datenquellenobjekt zu erleichtern. Bezieht sich der Kurzname beispielsweise auf eine Tabelle mit einem Index, so enthält der globale Katalog Informationen über den Index. Der Wrapper enthält die Zuordnungen zwischen den Datentypen von DB2® und den Datentypen der Datenquelle.

Momentan können bestimmte DB2 UDB-Dienstprogrammoperationen für Kurznamen nicht ausgeführt werden.

Sie können z. B. das Crossloader-Dienstprogramm nicht zum Ausführen von Ladeoperationen aus anderen Datenbanken mit Hilfe von Kurznamen verwenden.

### Zugehörige Konzepte:

- „Kurznamenspaltoptionen“ auf Seite 19

### Zugehörige Tasks:

- „Registrieren von Kurznamen für Datenquellen“ auf Seite 73

### Zugehörige Referenzen:

- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

- Anhang G, „Kurznamenspaltenoptionen für Systeme zusammengeschlüssener Datenbanken“, auf Seite 671
- Anhang F, „Kurznamenoptionen für Systeme zusammengeschlüssener Datenbanken“, auf Seite 661
- „Gültige Datenquellenobjekte“ auf Seite 18

## Gültige Datenquellenobjekte

Mit Kurznamen werden Objekte in der Datenquelle angegeben, auf die Sie zugreifen wollen. In der folgenden Tabelle sind die Objekttypen aufgelistet, für die Sie in einem System zusammengeschlüssener Datenbanken Kurznamen erstellen.

*Tabelle 3. Gültige Datenquellenobjekte*

Datenquelle	Gültige Objekte
DB2 für Linux, UNIX und Windows	Kurznamen, gespeicherte Abfragetabellen, Tabellen, Sichten
DB2 für z/OS und OS/390	Tabellen, Sichten
DB2 für iSeries	Tabellen, Sichten
DB2 für VM und VSE	Tabellen, Sichten
Informix	Tabellen, Sichten, Synonyme
Microsoft SQL Server	Tabellen, Sichten
ODBC	Tabellen, Sichten
Oracle	Tabellen, Sichten, Synonyme
Sybase	Tabellen, Sichten
Teradata	Tabellen, Sichten
BLAST	Für BLAST-Suchalgorithmen indizierte FASTA-Dateien
BioRS	BioRS-Datenbanken
Documentum	Objekte und registrierte Tabellen in einer Documentum-Docbase
Entrez	Entrez-Datenbank
Extended Search	Dateien von Datenquellen wie Lotus Notes-Datenbanken, Microsoft Access, Microsoft Index Server, Internetsuchmaschinen und LDAP-Verzeichnissen
HMMER	HMM-Datenbankdateien (Bibliotheken von Hierarchical Markov Models wie z. B. PFAM), die mit dem HMMER-Programm hmmpfam oder hmmsearch durchsucht werden können
Microsoft Excel	Dateien mit der Erweiterung .xls (Zugriff nur auf das erste Arbeitsblatt in der Arbeitsmappe)
Dateien mit Tabellenstruktur	Textdateien in einem bestimmten Format
WebSphere Business Integration-Adapter	WebSphere Business Integration-Geschäftsobjekte, die BAPIs in SAP, Geschäftskomponenten in Siebel und Komponentenschnittstellen in PeopleSoft zugeordnet sind

Tabelle 3. Gültige Datenquellenobjekte (Forts.)

Datenquelle	Gültige Objekte
Web-Services	Operationen in einer WSDL-Datei (WSDL = Web Services Description Language)
XML-Dateien	Elementgruppen in einem XML-Dokument

**Zugehörige Konzepte:**

- „Kurznamen und Datenquellenobjekte“ auf Seite 17
- „Kurznamenspaltenoptionen“ auf Seite 19

---

## Kurznamenspaltenoptionen

Sie können im globalen Katalog zusätzliche Metadaten über das Objekt mit dem Kurznamen bereitstellen. Diese Metadaten beschreiben Werte in bestimmten Spalten des Datenquellenobjekts. Sie werden Parametern zugeordnet, die als *Kurznamenspaltenoptionen* bezeichnet werden. Die Kurznamenspaltenoptionen bewirken, dass der Wrapper die Daten in einer Spalte auf andere Weise verarbeitet, als dies sonst der Fall wäre. Der SQL-Compiler und das Abfrageoptimierungsprogramm verwenden die Metadaten, um bessere Zugriffspläne für die Daten zu entwickeln.

Kurznamenspaltenoptionen werden auch dazu verwendet, dem Wrapper weitere Informationen zur Verfügung zu stellen. Bei XML-Datenquellen gibt die Kurznamenspaltenoption z. B. an, welchen XPath-Ausdruck der Wrapper verwenden muss, um die Spalte bei einer syntaktischen Analyse aus dem XML-Dokument zu entfernen.

Bei zusammengeschlossenen Datenbanken behandelt der DB2<sup>®</sup>-Server ein Datenquellenobjekt, auf das ein Kurzname verweist, als handelte es sich dabei um eine lokale DB2-Tabelle. Daher können Sie Kurznamenspaltenoptionen für alle Datenquellenobjekte setzen, für die Sie einen Kurznamen erstellen. Manche Kurznamenspaltenoptionen sind für spezielle Datenquellentypen bestimmt und können nur auf diese Datenquellen angewendet werden.

Angenommen, eine Datenquelle hat eine andere Sortierfolge als die zusammengeschlossene Datenbank. Normalerweise sortiert der Server mit zusammengeschlossenen Datenbanken Spalten mit Zeichendaten nicht in der Datenquelle. Stattdessen gibt er die Daten an die zusammengeschlossene Datenbank zurück und führt die Sortierung lokal aus. Nehmen Sie jedoch an, dass die Spalte ein Zeichendatentyp (CHAR oder VARCHAR) ist und ausschließlich numerische Zeichen ('0', '1', ..., '9') enthält. Diese Einstellung können Sie definieren, indem Sie der Kurznamenspaltenoption NUMERIC\_STRING den Wert 'Y' zuordnen. Auf diese Weise kann das Abfrageoptimierungsprogramm von DB2 die Sortierung in der Datenquelle durchführen. Wird die Sortierung fern durchgeführt, kann der Systemaufwand für die Übertragung der Daten zum Server mit zusammengeschlossenen Datenbanken und die lokale Sortierung vermieden werden.

Mit den Anweisungen ALTER NICKNAME können Sie Kurznamenspaltenoptionen für relationale Kurznamen definieren. Mit den Anweisungen CREATE NICKNAME und ALTER NICKNAME ist es möglich, Kurznamenspaltenoptionen für nicht relationale Kurznamen anzugeben.

**Zugehörige Konzepte:**

- „Datentypzuordnungen“ auf Seite 20

**Zugehörige Tasks:**

- „Arbeiten mit Kurznamen“ in *Systeme zusammengeschlossener Datenbanken*

**Zugehörige Referenzen:**

- Anhang G, „Kurznamenspaltenoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 671

---

## | Datentypzuordnungen

Die Datentypen der Datenquelle müssen den entsprechenden Datentypen von DB2<sup>®</sup> zugeordnet werden, sodass der Server mit zusammengeschlossenen Datenbanken die Daten von den Datenquellen abrufen kann. Im Folgenden sind einige Beispiele für Standarddatentypzuordnungen aufgeführt:

- Der Oracle-Typ FLOAT ist dem DB2-Typ DOUBLE zugeordnet.
- Der Oracle-Typ DATE ist dem DB2-Typ TIMESTAMP zugeordnet.
- Der Typ DATE von DB2 für z/OS<sup>™</sup> ist dem DB2-Typ DATE zugeordnet.

Für die meisten Datenquellen sind die Standardtypzuordnungen in den Wrappern enthalten. Die Standardtypzuordnungen für DB2-Datenquellen sind im DRDA<sup>®</sup>-Wrapper enthalten. Die Standardtypzuordnungen für Informix<sup>®</sup> sind im Informix-Wrapper enthalten, usw.

Für manche nicht relationale Datenquellen müssen die Datentypinformationen in der Anweisung CREATE NICKNAME angegeben werden. Die entsprechenden Datentypen von DB2 für Linux, UNIX<sup>®</sup> und Windows<sup>®</sup> müssen bei der Erstellung des Kurznamens für jede Spalte des Datenquellenobjekts angegeben werden. Jede Spalte muss einem bestimmten Feld oder einer bestimmten Spalte im Datenquellenobjekt zugeordnet werden.

Für relationale Datenquellen können die Standarddatentypzuordnungen außer Kraft gesetzt werden. Der Informix-Datentyp INTEGER wird z. B. standardmäßig dem DB2-Datentyp INTEGER zugeordnet. Sie können die Standardzuordnungen jedoch außer Kraft setzen und den Informix-Datentyp INTEGER dem DB2-Datentyp DECIMAL(10,0) zuordnen.

| Das Erstellen neuer Typzuordnungen oder das Ändern der Standardtypzuordnungen sollte vor der Erstellung von Kurznamen durchgeführt werden. Auf Kurznamen, die vor den Typzuordnungsänderungen erstellt wurden, haben die neuen Zuordnungen keine Auswirkung.

**Zugehörige Konzepte:**

- „Datentypzuordnungen in einem System zusammengeschlossener Datenbanken“ in *Systeme zusammengeschlossener Datenbanken*

---

## Funktionszuordnungen

Damit der Server mit zusammengeschlossenen Datenbanken eine Datenquellenfunktion erkennen kann, muss die Funktion einer gleichwertigen Funktion in DB2<sup>®</sup> für Linux, UNIX<sup>®</sup> und Windows<sup>®</sup> zugeordnet werden. DB2 Information Integrator stellt Standardzuordnungen zwischen vorhandenen integrierten Datenquellenfunktionen und den entsprechenden integrierten DB2-Funktionen bereit. Für die meisten Datenquellen sind die standardmäßigen Funktionszuordnungen in den Wrappern enthalten. Die Standardfunktionszuordnungen für Funktionen von DB2

für z/OS™ und OS/390® befinden sich im DRDA®-Wrapper. Die Standardfunktionszuordnungen zu Sybase-Funktionen sind im CTLIB-Wrapper enthalten, usw.

Für relationale Datenquellen können Sie eine Funktionszuordnung erstellen, wenn Sie eine Datenquellenfunktion verwenden wollen, die der Server mit zusammengeschlossenen Datenbanken nicht erkennt. Die Zuordnung, die Sie erstellen, besteht zwischen der Datenquellenfunktion und einer entsprechenden Funktion in der zusammengeschlossenen Datenbank. Funktionszuordnungen werden normalerweise verwendet, wenn eine neue integrierte oder benutzerdefinierte Funktion in der Datenquelle verfügbar wird. Funktionszuordnungen werden auch verwendet, wenn keine entsprechende Funktion in DB2 vorhanden ist. In diesem Fall muss auch eine Funktionsschablone erstellt werden.

**Zugehörige Konzepte:**

- „Funktionszuordnungen in einem System zusammengeschlossener Datenbanken“ in *Systeme zusammengeschlossener Datenbanken*
- „Indexspezifikationen“ auf Seite 21

---

## Indexspezifikationen

Beim Erstellen eines Kurznamens für eine Datenquellentabelle werden dem globalen Katalog Informationen über alle Indizes der Datenquellentabelle hinzugefügt. Das Abfrageoptimierungsprogramm verwendet diese Informationen zur Beschleunigung der Verarbeitung von verteilten Anforderungen. Die Kataloginformationen über den Index einer Datenquelle bestehen aus einem Satz von Metadaten und werden als *Indexspezifikation* bezeichnet. Ein Server mit zusammengeschlossenen Datenbanken erzeugt bei der Erstellung von Kurznamen für folgende Objekte keine Indexspezifikation:

- Eine Tabelle ohne Indizes
- Eine Sicht, für die normalerweise keine Indexinformationen im fernen Katalog gespeichert sind
- Ein Datenquellenobjekt, das über keinen fernen Katalog verfügt, von dem der Server mit zusammengeschlossenen Datenbanken die Indexinformationen abrufen kann

Angenommen, eine Tabelle erhält zusätzlich zu den zum Zeitpunkt der Erstellung des Kurznamens vorhandenen Indizes einen neuen Index. Da die Indexinformationen zum Zeitpunkt der Erstellung des Kurznamens an den globalen Katalog gegeben werden, bemerkt der Server mit zusammengeschlossenen Datenbanken den neuen Index nicht. Ebenso bemerkt der Server beim Erstellen eines Kurznamens für eine Sicht nicht die zugrunde liegende Tabelle (und deren Indizes), von der die Sicht erstellt wurde. Unter diesen Umständen können Sie dem globalen Katalog die erforderlichen Indexinformationen zur Verfügung stellen. Sie können eine Indexspezifikation für Tabellen ohne Indizes erstellen. Mit Hilfe der Indexspezifikation findet das Abfrageoptimierungsprogramm bei der Datensuche schnell die betreffenden Tabellenspalte(n).

**Zugehörige Konzepte:**

- „Indexspezifikationen in einem System zusammengeschlossener Datenbanken“ in *Systeme zusammengeschlossener Datenbanken*



---

## Interaktion mit einem System zusammenschlossener Datenbanken

Da es sich bei der zusammenschlossenen Datenbank um eine Datenbank von DB2® Universal Database handelt, können Sie zur Interaktion mit einem System zusammenschlossener Datenbanken wahlweise Folgendes verwenden:

- Den DB2-Befehlszeilenprozessor (CLP)
- Die GUI der DB2-Befehlszentrale
- Die GUI der DB2-Steuerzentrale
- Anwendungsprogramme
- DB2-Produktfamilie
- Web-Service-Provider

Die Schritte in der Dokumentation der zusammenschlossenen Datenbank stellen die Befehle und SQL-Anweisungen bereit, die im DB2-Befehlszeilenprozessor oder der GUI der DB2-Befehlszentrale eingegeben werden können. Wenn über die GUI der DB2-Steuerzentrale Tasks ausgeführt werden können, ist dies in der Dokumentation angegeben. Da die GUI der DB2-Steuerzentrale intuitiv ist, sind die Schritte zum Ausführen dieser Tasks über die DB2-Steuerzentrale in der vorliegenden Dokumentation nicht enthalten.

### DB2-Befehlszeilenprozessor (CLP)

Die meisten für die Installation, Konfiguration, Optimierung und Verwaltung des Systems zusammenschlossener Datenbanken erforderlichen Tasks können mit dem DB2-Befehlszeilenprozessor ausgeführt werden. In bestimmten Fällen müssen Sie entweder den DB2-Befehlszeilenprozessor oder die DB2-Befehlszentrale verwenden. Beispiele:

- Erstellen, Ändern oder Löschen benutzerdefinierter Datentypzuordnungen
- Erstellen, Ändern oder Löschen benutzerdefinierter Funktionszuordnungen

### DB2-Befehlszentrale

Mit der DB2-Befehlszentrale können verteilte Anforderungen erstellt und ausgeführt werden, ohne dass lange SQL-Anweisungen manuell eingegeben werden müssen. Verwenden Sie die DB2-Befehlszentrale, wenn Sie die Leistung des Systems zusammenschlossener Datenbanken optimieren wollen. Mit der DB2-Befehlszentrale kann die DB2-Funktionalität EXPLAIN auf einfache Weise verwendet werden, um die Zugriffspläne für verteilte Anforderungen anzuzeigen. Die DB2-Befehlszentrale kann auch zum Arbeiten mit dem Unterstützungstool für SQL verwendet werden.

### DB2-Steuerzentrale

Mit der GUI der DB2-Steuerzentrale können die meisten erforderlichen Tasks für die Installation, Konfiguration und Änderung des Systems zusammenschlossener Datenbanken ausgeführt werden. Die DB2-Steuerzentrale führt den Benutzer anhand von Anzeigen - Dialogfeldern und Assistenten - durch eine Task. Diese Anzeigen bieten interaktive Hilfe, wenn mit der Maus ein Bedienelement wie z. B. ein Listenfenster oder ein Befehlsdruckknopf berührt wird. Jede Anzeige verfügt zusätzlich über einen Hilfeknopf, der Informationen über die Task der Anzeige zur Verfügung stellt und eine Verbindung zu Informationen über zugehörige Konzepte und Referenzen herstellt.

Sie können für die Erstellung zusammenschlossener Objekte entweder einen Assistenten verwenden oder die Objekte einzeln erstellen.



Verwenden Sie zur Konfiguration des Zugriffs auf Web-Services-, WebSphere® Business Integration- und XML-Datenquellen die DB2-Steuerzentrale. Die in die DB2-Steuerzentrale integrierten Funktionen vereinfachen die Arbeitsschritte, die zur Konfiguration des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf diese Datenquellen erforderlich sind.

Die GUI der DB2-Steuerzentrale bietet die einfachste Möglichkeit, die folgenden grundlegenden Konfigurationstasks für die Datenquelle durchzuführen:

- Wrapper erstellen und die Wrapperoptionen festlegen
- Umgebungsvariablen der Datenquelle angeben
- Serverdefinitionen erstellen und Serveroptionen festlegen
- Benutzerzuordnungen erstellen und Benutzeroptionen festlegen
- Kurznamen erstellen und Kurznamen- oder Spaltenoptionen definieren

Nachdem Sie den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf die Datenquellen konfiguriert haben, können Sie die DB2-Steuerzentrale zur Ausführung folgender Arbeitsschritte verwenden:

- Ändern der Datenquellenkonfiguration
- Überwachen des Status von Kurznamen und Servern
- Verwalten der aktuellen Kurznamenstatistikdaten
- Erstellen und Ändern von Cachetabellen
- Angeben informativer Integritätsbedingungen für Kurznamen
- Erstellen ferner Tabellen über DB2 Information Integrator mit Hilfe von transparenter DDL

## Anwendungsprogramme

Anwendungen erfordern für die Arbeit mit Daten der zusammengeschlossenen Datenbank keine spezielle Codierung. Anwendungen greifen auf die gleiche Art und Weise auf das System zu wie jede andere beliebige DB2-Clientanwendung. Anwendungen verfügen über eine Schnittstelle zu der zusammengeschlossenen Datenbank, die sich auf dem Server befindet. Um Daten von Datenquellen abzurufen, übergeben Anwendungen Abfragen in DB2-SQL an die zusammengeschlossene Datenbank. DB2 Information Integrator verteilt die Abfragen dann an die entsprechenden Datenquellen, ruft die angeforderten Daten ab und gibt diese Daten an die Anwendungen zurück. Da die Interaktion von DB2 Information Integrator mit den Datenquellen jedoch über Kurznamen erfolgt, müssen folgende Punkte beachtet werden:

- Die SQL-Einschränkungen bei der Verwendung von Kurznamen
- Die Art und Weise der Ausführung von Operationen für Objekte mit Kurznamen

## DB2-Produktfamilie

Zur Interaktion mit einer zusammengeschlossenen Datenbank können Sie auch Host-Tools und Tools für mittlere Systeme verwenden. Hierzu gehören z. B. die folgenden Komponenten:

- DB2 SPUFI unter DB2 für z/OS™ und OS/390®
- Interactive SQL (STRSQL) unter DB2 für iSeries™

## Provider für Web-Services

Die Interaktion mit einer zusammenschlossenen Datenbank kann auch mit Hilfe des Web-Services-Wrappers über einen Web-Service-Provider ausgeführt werden.

### Zugehörige Konzepte:

- „Der Web-Services-Wrapper und das WSDL-Dokument“ auf Seite 433

### Zugehörige Tasks:

- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammenschlossenen Datenbanken“ auf Seite 439

---

## Kapitel 2. Planung für die Konfiguration zusammengesetzter Datenquellen

Die folgenden Abschnitte enthalten Informationen, die Ihnen bei der Planung Ihres Systems zusammengesetzter Datenbanken helfen sollen.

---

### Namenskonventionen für Objekte in zusammengesetzten Datenbanken

Wie auch bei anderen DB2-Objekten, gelten auch für Objekte zusammengesetzter Datenbanken (FDO = Federated Database Objects) bestimmten Benennungsregeln.

Es gibt folgende Objekte zusammengesetzter Datenbanken:

- Funktionszuordnungen
- Indexspezifikationen
- Kurznamen
- Server
- Typzuordnungen
- Benutzerzuordnungen
- Wrapper

Die Namen von Objekten zusammengesetzter Datenbanken müssen mit einem der folgenden Zeichen beginnen:

- Einem Buchstaben einschließlich aller zulässigen Buchstaben mit Akzent (z. B. Ö)
- Einem Mehrbytezeichen mit Ausnahme eines Mehrbyteleerzeichens (für Mehrbyteumgebungen)

Die Namen von Objekten zusammengesetzter Datenbanken dürfen nicht mit einer Zahl oder einem Unterstreichungszeichen beginnen.

An anderen Positionen dürfen sie die folgenden Zeichen enthalten:

- A bis Z
- 0 bis 9
- @, #, \$ und \_ (Unterstreichungszeichen)

Die Namen von Objekten zusammengesetzter Datenbanken dürfen maximal 128 Byte umfassen.

Optionen (z. B. Serveroptionen und Kurznamenoptionen) sowie Optionseinstellungen sind auf 255 Byte begrenzt.

Namen, die nicht in Anführungszeichen gesetzt sind, werden in Großschreibung umgesetzt.

#### Zugehörige Konzepte:

- „Benennungsregeln in einer NLS-Umgebung“ in *Systemverwaltung: Implementierung*
- „Benennungsregeln in einer Unicode-Umgebung“ in *Systemverwaltung: Implementierung*

#### Zugehörige Referenzen:

- „Prüfliste zur Planung der Konfiguration des Systems zusammengesetzter Datenbanken“ auf Seite 36
- „Beibehalten der von der Groß-/Kleinschreibung abhängigen Werte in einem System zusammengesetzter Datenbanken“ auf Seite 26

---

## Beibehalten der von der Groß-/Kleinschreibung abhängigen Werte in einem System zusammengesetzter Datenbanken

In einem System zusammengesetzter Datenbanken müssen in bestimmten Fällen Werte wie beispielsweise Benutzer-IDs und Kennwörter angegeben werden, für die in der Datenquelle die Groß-/Kleinschreibung berücksichtigt werden muss. Um sicherzustellen, dass die Groß-/Kleinschreibung korrekt definiert ist, wenn diese Werte an die Datenquelle übergeben werden, müssen die folgenden Richtlinien berücksichtigt werden:

- Geben Sie die Werte in der erforderlichen Groß-/Kleinschreibung an, und schließen Sie sie in die erforderlichen Anführungszeichen ein. Doppelte Anführungszeichen können bei Objektnamen wie z. B. dem Namen eines Wrappers oder bei Kurznamen optional verwendet werden. Einfache Anführungszeichen sind bei Optionswerten wie beispielsweise den Benutzerzuordnungsoptionen `REMOTE_AUTHID` und `REMOTE_PASSWORD` erforderlich.
- Bei Benutzer-IDs und Kennwörtern können Sie die Serveroptionen `FOLD_ID` und `FOLD_PW` definieren, um die Werte automatisch in die korrekte Groß-/Kleinschreibung umzusetzen. Wenn Sie mit dieser Option arbeiten, müssen Sie sich die jeweils erforderliche Groß-/Kleinschreibung für die verschiedenen Datenquellen nicht merken. Sie können die Werte in einer beliebigen Kombination aus Groß- und Kleinbuchstaben eingeben, denn sie werden vom System automatisch korrekt umgesetzt.

#### Von einer UNIX-Eingabeaufforderung des Betriebssystems:

Wenn Sie einen von der Groß-/Kleinschreibung abhängigen Wert in Anführungszeichen an der Eingabeaufforderung des Betriebssystems des Servers mit zusammengesetzten Datenbanken angeben, müssen Sie sicherstellen, dass die Anführungszeichen syntaktisch korrekt analysiert werden:

- Wenn die SQL-Anweisung doppelte Anführungszeichen, jedoch keine einfachen Anführungszeichen enthält, schließen Sie die gesamte Anweisung in einfache Anführungszeichen ein. Wenn Sie z. B. die SQL-Anweisung `CREATE NICKNAME my_nick FOR my_server."owner"."my_table"`

absetzen wollen, müssen Sie an der UNIX-Eingabeaufforderung folgenden Text eingeben:

```
DB2 'CREATE NICKNAME my_nick FOR my_server."owner"."my_table"'
```

- Wenn die SQL-Anweisung einfache Anführungszeichen, jedoch keine doppelten Anführungszeichen enthält, schließen Sie die gesamte Anweisung in doppelte Anführungszeichen ein.

Wenn Sie z. B. die SQL-Anweisung

```
CREATE USER MAPPING FOR USER SERVER my_server
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')
```

absetzen wollen, müssen Sie an der UNIX-Eingabeaufforderung folgenden Text eingeben:

```
DB2 "CREATE USER MAPPING FOR USER SERVER my_server
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')"
```

- Wenn die SQL-Anweisung sowohl einfache als auch doppelte Anführungszeichen enthält, gehen Sie wie folgt vor:
  - Schließen Sie die gesamte Anweisung in doppelte Anführungszeichen ein.
  - Geben Sie vor den Werten, die doppelte Anführungszeichen erfordern, einen Backslash ein.

Wenn Sie z. B. die SQL-Anweisung

```
CREATE USER MAPPING FOR "local_id" SERVER my_server
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')
```

absetzen wollen, müssen Sie an der UNIX-Eingabeaufforderung folgenden Text eingeben:

```
DB2 "CREATE USER MAPPING FOR \"local_id\" SERVER my_server
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')"
```

In den vorstehenden Beispielen wird davon ausgegangen, dass die SQL-Anweisungen über die UNIX-Eingabeaufforderung eingegeben werden und dass die Anweisung an den Befehl DB2 (ohne die Option -f) übergeben wird. Werden die SQL-Anweisungen über eine Datei eingegeben, wobei der Befehl DB2 mit der Option -f verwendet wird, müssen Sie die Anweisungen jeweils so eingeben, wie beim ersten Auftreten der einzelnen Beispiele gezeigt.

### Über eine Eingabeaufforderung eines Windows-Betriebssystems:

Um unter Windows die von Groß-/Kleinschreibung abhängigen Werte beizubehalten, muss jedem doppelten Anführungszeichen ein umgekehrter Schrägstrich vorangestellt werden. Beispiel: Für die Microsoft SQL Server-Tabelle *weekly\_salary* soll der Kurzname *NICK1* erstellt werden. Die Tabelle ist in der Datenbank *NORBASE* gespeichert. Das lokale Schema lautet *my\_schema*.

Geben Sie an der Windows-Eingabeaufforderung des Servers mit zusammengesetzten Datenbanken Folgendes ein:

```
DB2 CREATE NICKNAME nick1
  FOR norbase.\"my_schema\".\"weekly_salary\"
```

### Über die DB2-Eingabeaufforderung oder über ein Anwendungsprogramm:

Wenn Sie einen Wert über die DB2-Eingabeaufforderung oder über ein Anwendungsprogramm eingeben, können Sie die von der Groß-/Kleinschreibung abhängigen Werte beibehalten, indem Sie die Werte in die entsprechenden Anführungszeichen einschließen.

Beispiel: Sie wollen eine Benutzerzuordnung für die Benutzer-ID *local\_id* erstellen. Die ferne Benutzer-ID ist *my\_id* und das ferne Kennwort ist *my\_password*. Für alle drei Werte soll die Kleinschreibung beibehalten werden. Geben Sie hierzu an der DB2-Eingabeaufforderung Folgendes ein:

```
CREATE USER MAPPING FOR "local_id" SERVER my_server
  OPTIONS(REMOTE_AUTHID 'my_id', REMOTE_PASSWORD 'my_password')
```

#### Zugehörige Referenzen:

- Anhang D, „Serveroptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 641
- Anhang E, „Benutzerzuordnungsoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 659
- „Prüfliste zur Planung der Konfiguration des Systems zusammengeschlossener Datenbanken“ auf Seite 36

---

## Aktualisieren von Statistikdaten zur Datenquelle

Wenn Sie auf eine relationale Datenquelle zugreifen wollen, sollten Sie die Statistiken in der fernen Datenquelle aktualisieren, bevor Sie den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf die Datenquelle konfigurieren. Indem Sie sicherstellen, dass die ferne Datenquelle über aktuelle Statistiken verfügt, können Sie die Abfrageleistung verbessern.

Der Server mit zusammengeschlossenen Datenbanken verwendet zur Optimierung der Abfrageverarbeitung diejenigen Datenquellenstatistiken, die in der zusammengeschlossenen Datenbank gespeichert sind. Diese Statistiken werden gesammelt, wenn Sie einen Kurznamen für ein Datenquellenobjekt erstellen. Die zusammengeschlossene Datenbank stellt das Vorhandensein des Objekts in der Datenquelle sicher und versucht anschließend, bestehende statistische Daten für diese Datenquelle zu sammeln. Informationen, die für das Abfrageoptimierungsprogramm nützlich sind, werden aus den Katalogen der Datenquelle gelesen und dem Systemkatalog in der zusammengeschlossenen Datenbank hinzugefügt. Da einige oder alle der Kataloginformationen der Datenquelle vom Abfrageoptimierungsprogramm verwendet werden können, wird empfohlen, die Statistiken in der Datenquelle zu aktualisieren, bevor ein Kurzname erstellt wird. Verwenden Sie zum Aktualisieren der Datenquellenstatistiken den Befehl in der Datenquelle, der zum DB2-Befehl RUNSTATS äquivalent ist.

Die zusammengeschlossene Datenbank ruft diese statistischen Daten für ein Datenquellenobjekt ab, wenn Sie einen Kurznamen für das betreffende Objekt erstellen. Wenn die Datenquelle ihre Katalogstatistik für ein Objekt nach Erstellung des Kurznamens aktualisiert, werden die geänderten statistischen Informationen nicht an den Systemkatalog in der zusammengeschlossenen Datenbank weitergegeben. Um sicherzustellen, dass der Systemkatalog in der zusammengeschlossenen Datenbank die aktuellen Statistiken für das ferne Datenquellenobjekt widerspiegelt, müssen Sie anfordern, dass der Server mit zusammengeschlossenen Datenbanken die Statistiken aktualisiert.

**Aktion:** Ermitteln Sie die Datenquellenobjekte, auf die Sie zugreifen wollen. Für diese Objekte werden anschließend Kurznamen erstellt. Ermitteln Sie, für welche Datenquellen, zu denen diese Objekte gehören, Statistiken aktualisiert werden dürfen. Listen Sie diese Datenquellen in der Tabelle für die Datenquellenstatistik in der Planungsprüfliste auf.

#### Zugehörige Konzepte:

- „Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht“ in *Systeme zusammengeschlossener Datenbanken*

#### Zugehörige Referenzen:

- „Prüfliste zur Planung der Konfiguration des Systems zusammengeschlossener Datenbanken“ auf Seite 36

---

## Auswählen des korrekten Wrappers

Für den Zugriff auf die meisten Datenquellen können Sie lediglich einen Wrapper verwenden. Bei einigen Datenquellen können Sie jedoch auswählen, welchen Wrapper Sie für den Zugriff auf die Daten in diesen Datenquellen verwenden wollen.

Für den Zugriff auf Datenquellen, die die ODBC-API unterstützen, können Sie entweder die für diese Datenquellen entworfenen Wrapper oder den ODBC-Wrapper verwenden. Zu diesen Datenquellen gehören beispielsweise Oracle, Microsoft Excel und Microsoft SQL Server. Wenn Sie die Wrapper verwenden, die speziell für diese Datenquellen entworfen wurden, ist die Abfrageleistung normalerweise besser.

Verwenden Sie den ODBC-Wrapper, um auf alle Datenquellen zuzugreifen, die über einen ODBC-Treiber verfügen, aber nicht von spezifischen Datenquellenwrappern unterstützt werden, die mit DB2 Information Integrator bereitgestellt werden. Verwenden Sie beispielsweise den ODBC-Wrapper, um auf RedBrick-Datenquellen zuzugreifen.

### **Datenquellen von DB2 für Linux, UNIX und Windows**

Verwenden Sie den ODBC-Wrapper nicht, um auf Datenquellen von DB2 Universal Database für Linux, UNIX und Windows zuzugreifen. Der ODBC-Wrapper wird für den Zugriff auf Datenquellen von DB2 Universal Database für Linux, UNIX und Windows nicht unterstützt. Um auf Datenquellen von DB2 Universal Database für Linux, UNIX und Windows zuzugreifen, müssen Sie den DRDA-Wrapper verwenden.

### **Excel-Datenquellen**

Entsprechend Ihren Anforderungen können Sie für den Zugriff auf Excel-Daten anstelle des Excel-Wrappers auch den ODBC-Wrapper verwenden.

### **Informix-Datenquellen**

Verwenden Sie den ODBC-Wrapper nicht, um auf Datenquellen von Informix zuzugreifen. Der ODBC-Wrapper wird für den Zugriff auf Datenquellen von Informix nicht unterstützt. Um auf Datenquellen von Informix zuzugreifen, müssen Sie den Informix-Wrapper verwenden.

**Aktion:** Ermitteln Sie die Wrapper, die Sie für Ihr System zusammenschlossener Datenbanken erstellen wollen, und dokumentieren Sie diese in der Wrappertabelle in der Planungsprüfliste.

### **Zugehörige Konzepte:**

- „Methoden für den Zugriff auf Excel-Daten“ auf Seite 30

### **Zugehörige Tasks:**

- „Hinzufügen von Excel-Datenquellen zu einem Server mit zusammenschlossenen Datenbanken“ auf Seite 250
- „Hinzufügen von ODBC-Datenquellen zu einem Server mit zusammenschlossenen Datenbanken“ auf Seite 347

### **Zugehörige Referenzen:**

- „Prüfliste zur Planung der Konfiguration des Systems zusammenschlossener Datenbanken“ auf Seite 36



---

## Methoden für den Zugriff auf Excel-Daten

Für den Zugriff auf Daten in Microsoft® Excel-Arbeitsblättern können Sie entweder den Excel-Wrapper oder den ODBC-Wrapper verwenden.

Um Excel-Daten abfragen zu können, ist für beide Wrapper ein DB2®-Server mit zusammengeschlossenen Datenbanken erforderlich, der die Arbeitsblätter in Excel-Arbeitsmappen öffnen und lesen kann. Daher muss sich die Excel-Arbeitsmappe auf demselben Computer befinden wie der Server mit zusammengeschlossenen Datenbanken oder auf einem Netzlaufwerk, auf das Zugriff besteht.

Wenn Sie den Excel-Wrapper verwenden, muss die Excel-Anwendung auf dem Server mit zusammengeschlossenen Datenbanken installiert sein.

Wenn Sie den ODBC-Wrapper verwenden, muss sich der Excel ODBC-Treiber auf dem Server mit zusammengeschlossenen Datenbanken befinden. Dieser Treiber wird automatisch mit Microsoft Windows® installiert. Die Excel-Anwendung braucht nicht auf dem Server mit zusammengeschlossenen Datenbanken installiert zu sein.

Jeder Wrapper stellt bestimmte Anforderungen an die Position und das Layout der Daten in den Excel-Arbeitsmappen. Mit dem Excel-Wrapper kann lediglich auf Daten im ersten Arbeitsblatt einer Arbeitsmappe zugegriffen werden. Mit dem ODBC-Wrapper besteht Zugriff auf Daten in allen Arbeitsblättern einer Arbeitsmappe.

Die folgenden Beispiele zeigen, welche Anforderungen hinsichtlich des Layouts von Arbeitsblättern bei diesen beiden Wrappern bestehen.

### **Beispiel eines Arbeitsblatts mit Zeilen mit Kennsätzen und einer Formel:**

Dieses Beispiel zeigt ein Arbeitsblatt, das im oberen Teil mehrere Zeilen mit Kennsätzen sowie Leerzeilen und eine Formel in Zeile 13 enthält. Um auf die Daten in diesem Arbeitsblatt zuzugreifen, müssen Sie den Bereich der Zellen angeben, auf die Sie zugreifen wollen.

	A	B	C	D
1	<b>Compound Analysis</b>			
2				
3	<b>Compound Name</b>	<b>Weight</b>	<b>Molecular Count</b>	<b>Tested?</b>
4	compound_A	1.23	367	tested
5	compound_G		210	
6	compound_F	0.000425536	174	tested
7	compound_Y	1.000256		tested
8	compound_Q		1024	
9	compound_B	33.5362		
10	compound_S	0.96723	67	tested
11	compound_O	1.2		tested
12				
13	<b>Total Compounds Tested</b>			<b>5</b>

Abbildung 2. Arbeitsblatt mit mehreren Zeilen mit Kennsätzen und einer Formel

#### Bei Verwendung des Excel-Wrappers

Geben Sie in der Anweisung CREATE NICKNAME mit Hilfe der Option RANGE den Bereich der Zellen an. Schließen Sie nur die Daten im angegebenen Bereich ein. Schließen Sie in den Bereich keine Spaltenkennsätze ein. Zellen, die Formeln enthalten (wie beispielsweise SUM), geben das Ergebnis der Formel zurück und nicht die Formel selbst. Sofern Sie als Rückgabe nicht die Formelergebnisse wollen, schließen Sie in den Bereich keine Zellen ein, die Formeln enthalten. In diesem Beispiel lautet der in der Option RANGE angegebene Zellenbereich A4:D11.

#### Bei Verwendung des ODBC-Wrappers

Sie müssen für den Bereich der Zellen einen Namen erstellen, um die Position der Daten im Arbeitsblatt explizit anzugeben. Bei Excel wird dieser Zellenbereich als *benannter Bereich* bezeichnet. Der Excel ODBC-Treiber erkennt lediglich eine Zeile mit Kennsätzen, die erste Zeile im Bereich. Zwischen den Kennsätzen und den Daten sind keine Leerzeilen zulässig. Der benannte Bereich darf nur eine Reihe mit Spaltenkennsätzen enthalten. Der benannte Bereich wird in der Anweisung CREATE NICKNAME angegeben. Im benannten Bereich muss eine Reihe mit Spaltenkennsätzen enthalten sein. Enthält der benannte Bereich keine Zeile mit Spaltenkennsätzen, wird die erste Zeile mit Daten wie Spaltenkennsätze behandelt. Zellen, die Formeln enthalten (wie beispielsweise SUM), geben das Ergebnis der Formel zurück und nicht die Formel selbst. Sofern Sie als Rückgabe nicht die Formelergebnisse wollen, schließen Sie in den Bereich keine Zellen ein, die Formeln enthalten. In diesem Beispiel lautet der benannte Zellenbereich A3:D11.

### Beispiel eines Arbeitsblatts mit einer Zeile mit Kennsätzen:

Dieses Beispiel zeigt ein Arbeitsblatt, das nur eine Zeile mit Spaltenkennsätzen im oberen Teil enthält. Das Layout enthält keine zusätzlichen Zeilen mit Kennsätzen, Leerzeilen oder Zellen mit Formeln.

	A	B	C	D
1	Compound Name	Weight	Molecular Count	Tested?
2	compound_A	1.23	367	tested
3	compound_G		210	
4	compound_F	0.000425536	174	tested
5	compound_Y	1.000256		tested
6	compound_Q		1024	
7	compound_B	33.5362		
8	compound_S	0.96723	67	tested
9	compound_O	1.2		tested
10				
11				

Abbildung 3. Arbeitsblatt mit einer Zeile mit Spaltenkennsätzen in Zeile 1

#### Bei Verwendung des Excel-Wrappers

Geben Sie in der Anweisung CREATE NICKNAME mit Hilfe der Option RANGE den Bereich der Zellen an. Der Bereich darf nicht die Spaltenkennsätze in Zeile 1 enthalten. Der anzugebende Zellenbereich lautet daher A2:D9.

#### Bei Verwendung des ODBC-Wrappers

Für den Zugriff auf diese Daten braucht kein benannter Bereich erstellt zu werden. Der Name des Arbeitsblatts wird in der Anweisung CREATE NICKNAME angegeben. Der Wrapper liest die erste Zeile, die keine Leerzeile ist, als Kennsätze und verwendet die Informationen als Spaltennamen für den Kurznamen. Nachfolgende Zeilen werden als Daten gelesen.

### Beispiel eines Arbeitsblatt nur mit Daten:

Dieses Beispiel zeigt ein Arbeitsblatt, das nur Daten enthält. Es gibt keine Zeilen mit Spaltenkennsätzen, Leerzeilen und Zellen mit Formeln.

	compound_A	1.23	367	tested
	compound_G		210	
	compound_F	0.000425536	174	tested
	compound_Y	1.000256		tested
	compound_Q		1024	
	compound_B	33.5362		
	compound_S	0.96723	67	tested
	compound_O	1.2		tested

Abbildung 4. Arbeitsblatt nur mit Daten

### Bei Verwendung des Excel-Wrappers

Befinden sich die Daten im ersten Arbeitsblatt der Arbeitsmappe, greift der Wrapper auf die Daten zu, ohne die Option RANGE zu verwenden. Befinden sich die Daten hingegen in einem anderen Arbeitsblatt der Arbeitsmappe, müssen Sie die Option RANGE in der Anweisung CREATE NICK-NAME angeben.

### Bei Verwendung des ODBC-Wrappers

Wenn Sie für den Zugriff auf Excel-Daten den ODBC-Wrapper verwenden, ist der Wrapper auf die Funktionen beschränkt, die vom Excel ODBC-Treiber unterstützt werden. Der Excel ODBC-Treiber erfordert ein spezifisches Format für Arbeitsblätter. Der Treiber erwartet, dass die erste belegte Zeile die Spaltenkennsätze enthält. Enthält die erste belegte Zeile Daten, werden diese Daten wie Spaltenkennsätze für die übrigen Daten behandelt. Enthält das Arbeitsblatt keine Zeile mit Spaltenkennsätzen, werden die Daten in der ersten Zeile als Kennsätze verwendet und nicht als Daten. Somit geht die erste Zeile mit Daten verloren. Sie können dieses Problem lösen, indem Sie das Layout des betreffenden Arbeitsblatts ändern. Fügen Sie vor den Daten eine neue Zeile ein und fügen Sie für jede Datenspalte einen Kennsatz hinzu, sodass das Arbeitsblatt aussieht wie in dem Beispiel, das ein Zeile mit Kennsätzen enthält.

### Zugehörige Tasks:

- „Hinzufügen von Excel-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 250
- „Zugreifen auf Excel-Daten mit dem ODBC-Wrapper“ auf Seite 359

---

## Planen der Benutzerzuordnungen

Wenn ein Server mit zusammengeschlossenen Datenbanken eine Anforderung im Pushdown-Modus an eine Datenquelle senden muss, muss er zunächst eine Verbindung zu dieser Datenquelle aufbauen. Bei einigen Datenquellen verwendet der Server mit zusammengeschlossenen Datenbanken für den Verbindungsaufbau eine für diese Datenquelle gültige Benutzer-ID und ein gültiges Kennwort. Für diese Datenquellen müssen Sie eine Zuordnung zwischen der Benutzer-ID und dem Kennwort des Servers mit zusammengeschlossenen Datenbanken sowie der Benutzer-ID und dem Kennwort der Datenquelle definieren. Diese Zuordnung muss für alle Benutzer-IDs erstellt werden, die das System zusammengeschlossener Datenbanken zum Senden verteilter Anforderungen benutzen. Diese Zuordnung wird als *Benutzerzuordnung* bezeichnet.

Zum Erstellen einer Benutzerzuordnung für eine Gruppe von Benutzern, die mit der selben Benutzer-ID und dem selben Kennwort auf eine Datenquelle zugreifen, kann die DB2-Steuerzentrale verwendet werden.

**Aktion:** Stellen Sie die Benutzer-IDs fest, für die eine Benutzerzuordnung zwischen dem Server mit zusammengeschlossenen Datenbanken und der Datenquelle erstellt werden muss. Listen Sie die Benutzer-IDs des Servers mit zusammengeschlossenen Datenbanken und die zugehörigen Benutzer-IDs der Datenquellen in der Tabelle der Benutzerzuordnungen in der Planungsprüfliste auf.

### Zugehörige Referenzen:

- „Prüfliste zur Planung der Konfiguration des Systems zusammengeschlossener Datenbanken“ auf Seite 36

---

## Planen der Datentypzuordnungen

Die Datentypen von Datenquellen werden als *ferne* Datentypen bezeichnet, während Datentypen der zusammengeschlossenen Datenbank als *lokale* Datentypen bezeichnet werden.

Bei einigen Datenquellen enthalten die Wrapper die Standardzuordnungen zwischen den Datentypen der Datenquellen und den Datentypen der zusammengeschlossenen Datenbank. Beim Erstellen eines Kurznamens für ein Datenquellenobjekt werden Informationen zu den Spalten im Systemkatalog der zusammengeschlossenen Datenbank gespeichert. Die Datentypen für die Spalten stammen aus den standardmäßigen progressiven Datentypzuordnungen.

Bei anderen Datenquellen müssen Sie die Spalteninformationen und den Datentyp beim Erstellen des Kurznamens angeben. Einige der nicht relationalen Wrapper erstellen alle Spalten, die für den Zugriff auf eine Datenquelle erforderlich sind. Diese Spalten werden als *fixierte Spalten* bezeichnet. Bei anderen nicht relationalen Datenquellen können Sie einige oder alle Datentypen für die Spalten angeben.

Für Ihre Anwendungen können möglicherweise Datentypzuordnungen erforderlich sein, die sich von den Standardzuordnungen unterscheiden. Bei Wrappern, die das Angeben von Datentypzuordnungen zulassen, können Sie die Standardzuordnungen überschreiben, um Folgendes zu erreichen:

- Ändern einer Typzuordnung für alle Datenquellenobjekte, die sich auf einem bestimmten Server befinden
- Ändern einer Typzuordnung für ein bestimmtes Datenquellenobjekt

- Ändern einer Typzuordnung für einen bestimmten Datenquellentyp
- Ändern einer Typzuordnung für einen bestimmten Datenquellentyp und eine bestimmte Datenquellenversion

Verwenden Sie die Anweisung `CREATE TYPE MAPPING`, um neue Datentypzuordnungen zu definieren. Von Ihnen erstellte Zuordnungen werden in der Sicht `SYSCAT.TYPEMAPPINGS` im globalen Katalog der zusammengeschlossenen Datenbanken gespeichert.

Ändern Sie eine Datentypenzuordnung, *bevor* Sie für die Datenquellenobjekte Kurznamen erstellen. Wenn Sie einen Kurznamen für ein Datenquellenobjekt erstellen, füllt der Server mit zusammengeschlossenen Datenbanken den globalen Katalog mit den Informationen zur Tabelle. Diese Informationen sind u. a. der Kurzname, der Name der Datenquellentabelle, die Spaltennamen und die Datentypen, die für jede Tabellenspalte definiert wurden.

Nur Kurznamen, die nach der Änderung einer Zuordnung erstellt wurden, geben die neue Typenzuordnung wieder. Kurznamen, die vor der Änderung der Zuordnung erstellt wurden, verwenden die standardmäßige Datentypenzuordnung.

Werden die Datentypzuordnungen nach Erstellung der Kurznamen erstellt, müssen Sie alle Kurznamen ändern, um jeweils die neue Zuordnung wiederzugeben, oder die Kurznamen löschen und anschließend erneut erstellen.

**Hinweis:** Enthält eine Datenquellentabelle Spalten, die von den verfügbaren Datentypen abweichen oder vom Benutzer definiert wurden, können Sie eine der folgenden Vorgehensweisen auswählen:

- Sie können die Typenzuordnung in der zusammengeschlossenen Datenbank erstellen, bevor Sie einen Kurznamen für die entsprechende Datenquellentabelle generieren. Durch die Erstellung der Typenzuordnungen vor dem Generieren der Kurznamen erkennt der Server mit zusammengeschlossenen Datenbanken, welchen Datentyp er den Spalten zuordnen soll. Werden die Zuordnungen für diese abweichenden oder benutzerdefinierten Datentypen nicht vor der Eingabe der Anweisung `CREATE NICKNAME` erstellt, gibt das System eine Fehlermeldung aus.
- Wenn die Spalten in der Datenquellentabelle die folgenden Bedingungen erfüllen, gilt Folgendes:
  - Die Spalten stellen benutzerdefinierte Datentypen dar, die auf systemeigenen oder internen Datentypen basieren.
  - Die Spalten verfügen über Attribute, die für die Datentypzuordnungen nicht unterstützt werden.

Sie können eine Sicht für die Datenquelle erstellen, in der die Spalten dem zugrunde liegenden integrierten Datentyp zugeordnet oder in diesen *umgesetzt* werden. Anschließend können Sie einen Kurznamen für die Sicht anstatt für die Tabelle erstellen.

**Aktion:** Stellen Sie die Datentypzuordnungen fest, für die neue Zuordnungen definiert werden sollen. Listen Sie die zu erstellenden Datenquellen und Typenzuordnungen in der Tabelle der Datentypzuordnungen in der Planungsprüfliste auf.

**Zugehörige Konzepte:**

- „Datentypzuordnungen in einem System zusammengeschlossener Datenbanken“ in *Systeme zusammengeschlossener Datenbanken*

**Zugehörige Referenzen:**

- „Prüfliste zur Planung der Konfiguration des Systems zusammengeschlossener Datenbanken“ auf Seite 36
- „Unterstützte Datentypen für nicht relationale Datenquellen“ auf Seite 581

---

## Planen der Funktionszuordnungen

DB2 für UNIX und Windows stellt Standardfunktionszuordnungen zwischen vorhandenen integrierten Datenquellenfunktionen und integrierten DB2-Funktionen bereit. Für die meisten Datenquellen sind die Standardfunktionszuordnungen in den Wrappern enthalten. Bei einigen nichtrelationalen Datenquellen können die Standardfunktionszuordnungen nicht geändert werden.

Um eine Datenquellenfunktion verwenden zu können, die der Server mit zusammengeschlossenen Datenbanken nicht erkennt, müssen Sie eine Funktionszuordnung erstellen. Die Zuordnung, die Sie erstellen, besteht zwischen der Datenquellenfunktion und einer entsprechenden Funktion in der zusammengeschlossenen Datenbank. Funktionszuordnungen werden normalerweise verwendet, wenn eine neue integrierte oder benutzerdefinierte Funktion in der Datenquelle verfügbar wird. Funktionszuordnungen werden auch verwendet, wenn keine entsprechende Funktion in DB2 vorhanden ist. In diesem Fall müssen Sie vor der Erstellung der Funktionszuordnung eine Funktionsschablone in der zusammengeschlossenen Datenbank generieren.

**Aktion:** Stellen Sie fest, ob für die verwendeten Datenquellen Funktionszuordnungen erstellt werden müssen. Listen Sie die erforderlichen Funktionszuordnungen in der Funktionszuordnungstabelle in der Planungsprüfliste auf.

**Zugehörige Konzepte:**

- „Funktionszuordnungen“ auf Seite 20

---

## Prüfliste zur Planung der Konfiguration des Systems zusammengeschlossener Datenbanken

Die Konfiguration des Systems zusammengeschlossener Datenbanken kann durch die Verwendung der folgenden Planungsprüfliste vereinfacht werden. Anhand dieser Prüfliste werden Ihnen Methoden zur Konfigurationsoptimierung des Systems zusammengeschlossener Datenbanken aufgezeigt.

### Prüfliste zu den Namenskonventionen für Objekte zusammengeschlossener Datenbanken

Verfügen Sie über ausreichende Kenntnisse zu den Namenskonventionen für Objekte zusammengeschlossener Datenbanken? Unter den zugehörigen Referenzen am Ende des vorliegenden Abschnitts finden Sie Programmverbindungen (Links), über die Sie Informationen zu den Namenskonventionen für Objekte zusammengeschlossener Datenbanken lokalisieren können.



## Prüfliste zur Beibehaltung der von der Groß-/Kleinschreibung abhängigen Werte

Möchten Sie die Serveroptionen FOLD\_ID und FOLD\_PW verwenden, um die Groß-/Kleinschreibung in Benutzer-IDs und Kennwörtern beizubehalten, die an die verwendeten Datenquellen übertragen werden? Verwenden Sie die folgende Tabelle, um festzustellen, auf welche Serverdefinitionen diese Optionen angewendet werden müssen.

*Tabelle 4. Planungsprüfliste: Einstellungen der Serveroptionen FOLD\_ID und FOLD\_PW auf dem System zusammengeschlossener Datenbanken*

Datenquelle	Welchen Namen möchten Sie für den Server in der Serverdefinition für diese Datenquelle angeben?	Welche Einstellung möchten Sie für die Serveroption FOLD_ID angeben?	Welche Einstellung möchten Sie für die Serveroption FOLD_PW angeben?

## Prüfliste zur Datenquellenstatistik

In der folgenden Tabelle sollten Sie die Datenquellen dokumentieren, die zum System zusammengeschlossener Datenbanken gehören. Geben Sie die Datenquellen an, für die Sie die Statistik aktualisieren möchten, bevor Sie den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf die Datenquelle konfigurieren. Als Beispiel ist in dieser Tabelle DB2 UDB für Linux, UNIX und Windows aufgelistet.

*Tabelle 5. Planungsprüfliste: Statistikdaten für Datenquellen, die im System zusammengeschlossener Datenbanken aktualisiert werden müssen*

Datenquelle	Werden von dieser Datenquelle Kataloginformationen verwaltet? (J/N)	Möchten Sie für diese Datenquelle Statistikdaten aktualisieren? (J/N)	Name des Dienstprogramms, mit dem die Datenquelle die Statistikdaten aktualisiert
DB2 für Linux, UNIX und Windows	J	J	RUNSTATS

## Prüfliste zu Datentypzuordnungen

Dokumentieren Sie in der folgenden Tabelle die Datentypen der Datenquellen und die entsprechenden Datentypen des Servers mit zusammengeschlossenen Datenbanken, für die eine Zuordnung erstellt werden muss.

*Tabelle 6. Planungsprüfliste: Für das System zusammengeschlossener Datenbanken zu erstellende Datentypzuordnungen*

Datenquelle	Welchen Namen möchten Sie für den Server in der Serverdefinition für diese Datenquelle angeben?	Datentyp der Datenquelle	Datentyp des Servers mit zusammengeschlossenen Datenbanken

## Prüfliste zu Benutzerzuordnungen

Dokumentieren Sie in der folgenden Tabelle die Benutzer-IDs des Servers mit zusammengeschlossenen Datenbanken und die entsprechenden Benutzer-IDs aller Datenquellen, die zum System zusammengeschlossener Datenbanken gehören.

*Tabelle 7. Planungsprüfliste: Für das System zusammengeschlossener Datenbanken zu erstellende Benutzerzuordnungen*

		Datenquelle	Datenquelle	Datenquelle
		_____	_____	_____
Benutzername	Benutzer-ID für DB2 für Linux, UNIX und Windows	Benutzer-ID	Benutzer-ID	Benutzer-ID

## Prüfliste für Wrapper

Dokumentieren Sie in der folgenden Tabelle die Wrapper, die Sie erstellen möchten.

*Tabelle 8. Planungsprüfliste: Für das System zusammengeschlossener Datenbanken zu erstellende Wrapper*

Datenquelle	Standard-wrappername	Vorgesehener Name für den Wrapper
BioRS	keiner	
BLAST	keiner	
Geschäftsanwendungen (Wrapper für WebSphere Business Integration)	keiner	
DB2 Universal Database™ für Linux, UNIX und Windows®	DRDA	
DB2 Universal Database für z/OS und OS/390®		
DB2 Universal Database für iSeries		
DB2 Server für VM und VSE		
Documentum	keiner	
Entrez	keiner	
Excel	keiner	
Extended Search	keiner	
HMMER	keiner	
Informix	INFORMIX	
Microsoft® SQL Server	MSSQLODBC3	
Oracle	NET8	
ODBC	keiner	
OLE DB	OLEDB	
Sybase	CTLIB	
Dateien mit Tabellenstruktur	keiner	
Teradata	TERADATA	
Web-Services	keiner	
XML	keiner	

**Zugehörige Konzepte:**

- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63
- „Methoden für den Zugriff auf Excel-Daten“ auf Seite 30

**Zugehörige Tasks:**

- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43
- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58

**Zugehörige Referenzen:**

- „Überlegungen zu Landessprachen in der zusammengeschlossenen Datenbank“ auf Seite 51
- „Namenskonventionen für Objekte in zusammengeschlossenen Datenbanken“ auf Seite 25
- „Beibehalten der von der Groß-/Kleinschreibung abhängigen Werte in einem System zusammengeschlossener Datenbanken“ auf Seite 26
- „Aktualisieren von Statistikdaten zur Datenquelle“ auf Seite 28
- „Planen der Datentypzuordnungen“ auf Seite 34
- „Planen der Funktionszuordnungen“ auf Seite 36
- „Planen der Benutzerzuordnungen“ auf Seite 34
- „Auswählen des korrekten Wrappers“ auf Seite 29

---

## **Teil 2. Server mit zusammengeschlossenen Datenbanken und zusammengeschlossene Datenbank**



---

## Kapitel 3. Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken

---

### Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken

Durch Überprüfen einiger wichtiger Einstellungen auf dem Server mit zusammengeschlossenen Datenbanken lassen sich mögliche Konfigurationsprobleme vermeiden.

#### Vorgehensweise:

Um die Konfiguration des Servers mit zusammengeschlossenen Datenbanken zu prüfen, gehen Sie wie folgt vor:

- Stellen Sie sicher, dass eine Programmverbindung (Link-edit) zwischen den Wrapperbibliotheksdateien und der Client-Software der Datenquellen besteht (UNIX).
- Überprüfen Sie, ob der Parameter FEDERATED auf YES (ja) gesetzt ist.

Nachdem Sie die Konfiguration des Servers mit zusammengeschlossenen Datenbanken geprüft haben, müssen Sie eine zusammengeschlossene Datenbank erstellen.

#### Zugehörige Tasks:

- „Bestätigen der Programmverbindung (Link-edit) zu den Wrapperbibliotheksdateien (UNIX)“ auf Seite 43
- „Überprüfung des Parameters FEDERATED“ auf Seite 50
- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58

---

### Bestätigen der Programmverbindung (Link-edit) zu den Wrapperbibliotheksdateien (UNIX)

#### Bestätigen der Programmverbindung (Link-edit) zu den Wrapperbibliotheksdateien (UNIX)

Bei der Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken muss sichergestellt werden, dass eine Programmverbindung (Link-edit) zu den Wrapperbibliotheksdateien besteht.

Auf UNIX-Servern mit zusammengeschlossenen Datenbanken muss für einige Wrapper eine Programmverbindung zur Client-Software der entsprechenden Datenquellen hergestellt werden. Beim Installieren von DB2 Information Integrator wird versucht, eine solche Programmverbindung herzustellen. Beim Herstellen der Programmverbindung wird eine Wrapperbibliothek für jede Datenquelle erstellt, mit der der Server mit zusammengeschlossenen Datenbanken kommunizieren soll.

Diese Task gilt nur für die folgenden Datenquellen:

- Informix



- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

Bevor Sie den Server mit zusammengeschlossenen Datenbanken und die zusammengeschlossene Datenbank für den Zugriff auf Datenquellen konfigurieren, sollten Sie sicherstellen, dass die Programmverbindung zu den Wrapperbibliotheksdateien erfolgreich hergestellt wurde.

#### **Voraussetzungen:**

Ein Server mit zusammengeschlossenen Datenbanken, der ordnungsgemäß für den Zugriff auf Datenquelle eingerichtet ist. Dies umfasst die Installation und Konfiguration der erforderlichen Software wie beispielsweise der Client-Software von Datenquellen.

#### **Vorgehensweise:**

Ermitteln Sie den Status der Programmverbindung zu den Wrapperbibliotheksdateien wie folgt:

- War die Herstellung der Programmverbindung erfolgreich, wird die Wrapperbibliotheksdatei in dem Verzeichnis angezeigt, in dem DB2 Information Integrator installiert ist.
- Ist die Herstellung der Programmverbindung fehlgeschlagen, überprüfen Sie die Fehlnachrichtendatei in dem Verzeichnis, in dem DB2 Information Integrator installiert ist.
- Wurde die Herstellung der Programmverbindung nicht ausgeführt, enthält das Verzeichnis, in dem DB2 Information Integrator installiert ist, weder die Bibliotheksdatei noch die Nachrichtendatei. Sie müssen die Prozedur zum Herstellen der Programmverbindung manuell ausführen.

#### **Zugehörige Tasks:**

- „Überprüfung der Wrapperbibliotheksdateien (UNIX)“ auf Seite 44
- „Überprüfung der Nachrichtendateien für das Herstellen von Programmverbindungen (UNIX)“ auf Seite 46
- „Manuelles Herstellen einer Programmverbindung zwischen den Wrapperbibliotheken und der Client-Software von Datenquellen“ auf Seite 47
- „Hinzufügen von Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken mit Hilfe der DB2 UDB-Steuerzentrale“ auf Seite 64

## **Überprüfung der Wrapperbibliotheksdateien (UNIX)**

Bei der Sicherstellung, dass eine Programmverbindung (Link-edit) zwischen den Wrapperbibliotheken und der Client-Software von Datenquellen besteht, müssen auch die Wrapperbibliotheksdateien überprüft werden.

Die Wrapperbibliotheksdateien sind für den Zugriff auf die Datenquellen erforderlich. Bei einigen Datenquellen werden die Bibliotheksdateien beim Installieren von DB2 Information Integrator dem Server mit zusammengeschlossenen Datenbanken hinzugefügt. Bei anderen Datenquellen muss eine Prozedur zur Herstellung der Programmverbindung ausgeführt werden, um die Bibliotheksdateien zu erstellen.

Diese Task gilt nur für die folgenden Datenquellen:

- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

#### **Vorgehensweise:**

Um zu überprüfen, ob sich die Wrapperbibliotheksdateien auf dem Server mit zusammengeschlossenen Datenbanken befinden, gehen Sie wie folgt vor:

1. Suchen Sie nach den Bibliotheksdateien im Verzeichnispfad der Wrapperbibliothek. Sie müssen sicherstellen, dass für jede Datenquelle, auf die Sie zugreifen wollen, jeweils eine Bibliotheksdatei auf dem Server mit zusammengeschlossenen Datenbanken vorhanden ist.
2. Enthält das Verzeichnis keine Bibliotheksdateien, müssen Sie manuell eine Programmverbindung (Link) zwischen den Wrapperbibliotheken und der Client-Software der Datenquellen herstellen.

#### **Zugehörige Tasks:**

- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43
- „Überprüfung der Nachrichtendateien für das Herstellen von Programmverbindungen (UNIX)“ auf Seite 46
- „Manuelles Herstellen einer Programmverbindung zwischen den Wrapperbibliotheken und der Client-Software von Datenquellen“ auf Seite 47

#### **Zugehörige Referenzen:**

- „Wrapperbibliotheksdateien“ auf Seite 45
- „Bibliotheksdateien der benutzerdefinierten Life Sciences-Funktionen“ auf Seite 503

## **Wrapperbibliotheksdateien**

Die Wrapperbibliotheksdateien sind für den Zugriff auf die Datenquellen erforderlich. Bei einigen Datenquellen werden die Bibliotheksdateien beim Installieren von DB2 Information Integrator dem Server mit zusammengeschlossenen Datenbanken hinzugefügt. Bei anderen Datenquellen muss eine Prozedur für das Herstellen der Programmverbindung ausgeführt werden, um die Bibliotheksdateien zu erstellen.

Die Wrapperbibliotheksdateien sind erforderlich, wenn Sie den Wrapper für die Datenquelle registrieren.

Sie sollten überprüfen, ob sich die Wrapperbibliotheksdateien auf dem Server mit zusammengeschlossenen Datenbanken befinden. Für jede Datenquelle, auf die Sie zugreifen wollen, muss jeweils ein Gruppe von Wrapperbibliotheksdateien vorhanden sein.

Befinden sich die Wrapperbibliotheksdateien nicht auf dem Server mit zusammengeschlossenen Datenbanken, müssen Sie die Prozedur für das Herstellen einer Programmverbindung manuell ausführen, um die Bibliotheksdateien zu erstellen.

### Zugehörige Tasks:

- „Überprüfung der Wrapperbibliotheksdateien (UNIX)“ auf Seite 44
- „Manuelles Herstellen einer Programmverbindung zwischen den Wrapperbibliotheken und der Client-Software von Datenquellen“ auf Seite 47

### Zugehörige Referenzen:

- „Bibliotheksdateien des BLAST-Wrappers“ auf Seite 122
- „Bibliotheksdateien für den BioRS-Wrapper“ auf Seite 82
- „Bibliotheksdateien des DB2-Wrappers“ auf Seite 183
- „Bibliotheksdateien für den Documentum-Wrapper“ auf Seite 197
- „Bibliotheksdateien des Entrez-Wrappers“ auf Seite 225
- „Bibliotheksdateien für den Excel-Wrapper“ auf Seite 252
- „Bibliotheksdateien für den Extended Search-Wrapper“ auf Seite 272
- „Bibliotheksdateien für den HMMER-Wrapper“ auf Seite 299
- „Bibliotheksdateien für den Informix-Wrapper“ auf Seite 320
- „Bibliotheksdateien des Microsoft SQL Server-Wrappers“ auf Seite 337
- „ODBC-Wrapper-Bibliotheksdateien“ auf Seite 351
- „OLE DB-Wrapper-Bibliotheksdateien“ auf Seite 367
- „Oracle-Wrapper-Bibliotheksdateien“ auf Seite 378
- „Sybase-Wrapper-Bibliotheksdateien“ auf Seite 393
- „Teradata-Wrapper-Bibliotheksdateien“ auf Seite 422
- „Bibliotheksdateien des Wrappers für Dateien mit Tabellenstruktur“ auf Seite 407
- „Bibliotheksdateien des WebSphere Business Integration-Wrappers“ auf Seite 146
- „Bibliotheksdateien des Web-Services-Wrappers“ auf Seite 441
- „XML-Wrapper-Bibliotheksdateien“ auf Seite 476

## Überprüfung der Nachrichtendateien für das Herstellen von Programmverbindungen (UNIX)

Bei der Sicherstellung, dass eine Programmverbindung (Link-edit) zwischen den Wrapperbibliotheken und der Client-Software von Datenquellen besteht, müssen auch die Nachrichtendateien für das Herstellen von Programmverbindungen überprüft werden.

Schlägt die Herstellung einer Programmverbindung fehl, werden entsprechende Fehlermeldungen in der Nachrichtendatei im Bibliotheksverzeichnis aufgeführt. Das Vorhandensein einer Nachrichtendatei allein bedeutet nicht, dass die Herstellung der Programmverbindung fehlgeschlagen ist. Auch wenn die Programmverbindung erfolgreich hergestellt werden konnte, befindet sich im Bibliotheksverzeichnis eine Nachrichtendatei.

Sie müssen die Nachrichtendatei öffnen, um zu ermitteln, ob die Herstellung der Programmverbindung fehlgeschlagen ist.

Diese Task ist nur für die folgenden Datenquellen erforderlich:

- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

### Vorgehensweise:

Um zu ermitteln, ob die Herstellung der Programmverbindung fehlgeschlagen ist, öffnen Sie die Nachrichtendateien für das Herstellen von Programmverbindungen. Diese Nachrichtendateien befinden sich im Unterverzeichnis `lib` oder `lib64` des Verzeichnisses, in dem DB2 installiert ist. Die nachstehende Tabelle enthält eine Liste der Namen der Nachrichtendateien für das Herstellen von Programmverbindungen nach Datenquellen.

*Tabelle 9. Namen der Nachrichtendateien für das Herstellen von Programmverbindungen nach Datenquellen*

Datenquelle	Nachrichtendateinamen
Informix	djxlinkInformix.out
Microsoft SQL Server	djxlinkMssql.out
Oracle	djxlinkOracle.out
Sybase	djxlinkSybase.out
Teradata	djxlinkTeradata.out

Für das Fehlschlagen von Programmverbindungen bei der Konfiguration des Servers mit zusammengeschlossenen Datenbanken kann es mehrere Gründe geben:

- Wird die Client-Software der Datenquelle nicht installiert, bevor der Versuch unternommen wird, eine Programmverbindung herzustellen, schlägt die Herstellung der Programmverbindung fehl. Wenn Sie beispielsweise die Informix-Client-Software nicht vor der Installation der DB2-Server-Software installieren, schlägt die Herstellung der Programmverbindung fehl. Ebenso schlägt die Herstellung der Programmverbindung fehl, wenn Sie die Sybase Open Client-Software nicht vor der Installation von DB2 Information Integrator installieren. In diesen Fällen müssen Sie die Programmverbindung manuell herstellen.
- Prüfen Sie, ob die Version der Client-Software der Datenquelle unterstützt wird. Wird die installierte Version der Client-Software nicht unterstützt, schlägt die Herstellung der Programmverbindung fehl. In diesem Fall müssen Sie zunächst eine Clientversion installieren, die unterstützt wird, und die Programmverbindung anschließend manuell herstellen.

### Zugehörige Tasks:

- „Bestätigen der Programmverbindung (Link-edit) zu den Wrapperbibliotheksdateien (UNIX)“ auf Seite 43
- „Überprüfung der Wrapperbibliotheksdateien (UNIX)“ auf Seite 44
- „Manuelles Herstellen einer Programmverbindung zwischen den Wrapperbibliotheken und der Client-Software von Datenquellen“ auf Seite 47

## Manuelles Herstellen einer Programmverbindung zwischen den Wrapperbibliotheken und der Client-Software von Datenquellen

Bei der Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken muss unter Umständen auch manuell eine Programmverbindung zwischen den Wrapperbibliotheken und der Client-Software von Datenquellen hergestellt werden.

Enthält der Verzeichnispfad keine Bibliotheksdateien, müssen Sie manuell eine Programmverbindung (Link) zu den Wrapperbibliotheken herstellen.

Diese Task gilt nur für die folgenden Datenquellen:

- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata

#### **Voraussetzungen:**

Zum Ausführen der Prozeduren für das Herstellen einer Programmverbindung benötigen Sie Rootberechtigung.

Es muss mindestens ein DB2-Exemplar vorhanden sein, damit die Nachrichten der djxlinkxxx-Prozeduren in Ihrer Landessprache ausgegeben werden. Ist kein Exemplar vorhanden, können die Prozeduren zwar ausgeführt werden, doch werden in diesem Fall Fehlermeldungen angezeigt. Jede Fehlermeldung beginnt mit db2djxmsg: Error retrieving message number (Fehler beim Abrufen der Nachrichtennummer). Auf diese Fehlermeldung folgt eine Nachricht in englischer Sprache. Beispiel:

```
db2djxmsg: Error retrieving message number 9004
  (return code -2029059891 from sqlogmsg).
Begin processing for wrapper: INFORMIX
INFORMIXDIR = /wsdb/v82/bldsupp/AIX/informix2.81
db2djxmsg: Error retrieving message number 9015
  (return code -2029059891 from sqlogmsg).
Library libdb2informixF.a was built successfully.
db2djxmsg: Error retrieving message number 9006
  (return code -2029059891 from sqlogmsg).
End processing for wrapper: INFORMIX
```

#### **Vorgehensweise:**

Um schnell eine Programmverbindung zwischen den Wrapperbibliotheken und der Client-Software von Datenquellen herzustellen, gehen Sie wie folgt vor:

1. Installieren und konfigurieren Sie gegebenenfalls die Client-Software auf dem DB2-Server mit zusammengeschlossenen Datenbanken.
2. Verwenden Sie die Produkt-CDs, um eine Programmverbindung herzustellen:
  - Bei Informix-Datenquellen: Führen Sie die DB2-Serverinstallation erneut aus und geben Sie hierbei die Installationsoption 'Standard' an.
  - Bei Microsoft SQL Server-, Oracle-, Sybase- und Teradata-Datenquellen: Führen Sie die Installation von DB2 Information Integrator erneut aus. Klicken Sie über die Klickstartleiste **Produkte installieren** an und befolgen Sie die Anweisungen des Assistenten.
3. Überprüfen Sie nach Herstellung der Programmverbindung die Berechtigungen für die Wrapperbibliotheken. Stellen Sie sicher, dass die Bibliotheken von den DB2-Exemplaren gelesen und ausgeführt werden können.

Als Alternative haben Sie die Möglichkeit, die Prozeduren für das Herstellen einer Programmverbindung wie folgt über die UNIX-Eingabeaufforderung auszuführen.

1. Öffnen Sie eine UNIX-Eingabeaufforderung.

- Führen Sie die Prozedur für das Herstellen einer Programmverbindung für jede Datenquelle aus, auf die Sie zugreifen wollen.

Die nachstehende Tabelle enthält eine Liste der Namen dieser Prozeduren nach Datenquellen geordnet.

*Tabelle 10. Prozeduren für das Herstellen einer Programmverbindung nach Datenquellen*

Datenquelle	Prozedurname
Informix	djxlinkInformix
Microsoft SQL Server	djxlinkMssql
Oracle	djxlinkOracle
Sybase	djxlinkSybase
Teradata	djxlinkTeradata

Wenn Sie beispielsweise den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Informix-Datenquellen konfigurieren, führen Sie über die UNIX-Eingabeaufforderung die Prozedur `djxlinkInformix` aus:

```
djxlinkInformix
```

- Setzen Sie für jedes DB2-Exemplar den Befehl **db2iupdt** ab, um den zusammengeschlossenen Zugriff auf die Datenquellen zu aktivieren.
- Überprüfen Sie nach Herstellung der Programmverbindung die Berechtigungen für die Wrapperbibliotheken. Stellen Sie sicher, dass die Bibliotheken von den DB2-Exemplareignern gelesen und ausgeführt werden können.

**Achtung:** Es gibt eine weitere Prozedur (`djxlink`), die versucht, für alle von DB2 Information Integrator unterstützte Datenquellen jeweils eine Wrapperbibliothek zu erstellen. Wenn Sie die Prozedur `djxlink` ausführen, die Client-Software jedoch nur für einige der Datenquellen installiert ist, wird für jede nicht installierte Datenquelle jeweils eine Fehlermeldung angezeigt.

#### **Zugehörige Tasks:**

- „Installieren von DB2 Information Integrator (Windows)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Installieren von DB2 Information Integrator (UNIX)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43
- „Überprüfung des Parameters FEDERATED“ auf Seite 50
- „Überprüfung der Nachrichtendateien für das Herstellen von Programmverbindungen (UNIX)“ auf Seite 46

---

## Überprüfung des Parameters FEDERATED

Bei der Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken muss auch der Parameter FEDERATED geprüft werden.

Bevor Sie dem Server mit zusammengeschlossenen Datenbanken und der zusammengeschlossenen Datenbank Datenquellen hinzufügen, sollten Sie die Einstellung des Parameters FEDERATED überprüfen.

Der Parameter FEDERATED muss auf YES (Ja) gesetzt sein, damit der Server mit zusammengeschlossenen Datenbanken auf die Datenquellen zugreifen kann.

### Voraussetzungen:

- DB2 Information Integrator muss auf einem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.

### Vorgehensweise:

Um die Einstellung des Parameters FEDERATED zu überprüfen, gehen Sie wie folgt vor:

1. Setzen Sie den nachstehenden DB2-Befehl ab, um alle Parameter sowie deren aktuelle Einstellungen anzuzeigen:  
`GET DATABASE MANAGER CONFIGURATION`
2. Überprüfen Sie die Einstellung des Parameters CONCENTRATOR. Der Parameter CONCENTRATOR und der Parameter FEDERATED dürfen nicht gleichzeitig auf YES (Ja) gesetzt sein. Ist der Parameter CONCENTRATOR auf YES gesetzt, ändern Sie die Einstellung in NO (Nein). Setzen Sie den folgenden DB2-Befehl ab, um die Einstellung zu ändern:  
`UPDATE DATABASE MANAGER CONFIGURATION USING CONCENTRATOR NO`
3. Überprüfen Sie die Einstellung des Parameters FEDERATED. Ist der Parameter FEDERATED auf NO gesetzt, ändern Sie die Einstellung in YES. Setzen Sie den folgenden DB2-Befehl ab, um die Einstellung zu ändern:  
`UPDATE DATABASE MANAGER CONFIGURATION USING FEDERATED YES`

### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

### Zugehörige Tasks:

- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43



---

## Kapitel 4. Erstellen einer zusammengeschlossenen Datenbank

Bevor Sie den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Ihre Datenquellen konfigurieren können, müssen Sie eine Datenbank erstellen, die als zusammengeschlossene Datenbank verwendet wird.

---

### Überlegungen zu Landessprachen in der zusammengeschlossenen Datenbank

Bei vielen relationalen Datenquellen führt der Wrapper beim Herstellen einer Verbindung zur Datenquelle folgende Tasks aus:

1. Ermitteln der Codepage und des Gebiets der zusammengeschlossenen Datenbank.
2. Zuordnen der Codepage und des Gebiets zu einem Ländereinstellungsnamen des Datenquellenclients.
3. In Abhängigkeit von der Datenquelle Setzen einer Umgebungsvariable oder Aufrufen einer Datenquellen-API, um der Datenquelle Informationen zur Ländereinstellung des Clients zu übermitteln.
4. Die Datenquelle konvertiert dann die Zeichendaten zwischen der Codepage der fernen Datenbank und der Codepage der zusammengeschlossenen Datenbank.

Wenn die zusammengeschlossene Datenbank z. B. die Codepage 819 und das Gebiet US verwendet, ist die äquivalente Ländereinstellung des Oracle-Clients `American_America.WE8ISO8859P1`. Der Wrapper setzt dann die Variable `NLS_LANG` auf den Wert für die Ländereinstellung des Oracle-Clients. Wenn Daten von der Oracle-Datenbank an den Wrapper gesendet werden, konvertiert die Oracle-Datenbank die Daten vom codierten Zeichensatz `American_America.WE8ISO8859P1` in Codepage 819. Werden Daten von der Oracle-Datenbank an den Wrapper gesendet, konvertiert der Oracle-Server bzw. -Client die Daten von der Codepage der Oracle-Datenbank in die Codepage 819. Werden Daten vom Wrapper an die Oracle-Datenbank gesendet, konvertiert der Oracle-Server bzw. -Client die Daten von Codepage 819 in die Codepage der Oracle-Datenbank.

Bei relationalen Datenquellen, die keine Codepagekonvertierung ausführen, stellen einige der Wrapper die Ausführung der Konvertierung zur Verfügung.

Wird für die zusammengeschlossene Datenbank die Verwendung desselben codierten Zeichensatzes, desselben Gebiets und derselben Sortierfolge wie für die Datenquelle definiert, kann hierdurch die Leistung verbessert werden. Wenn für die zusammengeschlossene Datenbank und die Datenquelle die Verwendung desselben codierten Zeichensatzes, desselben Gebiets und derselben Sortierfolge definiert werden, ist die Codepagekonvertierung nicht erforderlich. Die Verwendung derselben landessprachlichen Einstellungen kann bei der Übertragung großer Mengen von Zeichendaten die Leistung verbessern.



Um den codierten Zeichensatz und das Gebiet für die zusammengeschlossene Datenbank anzugeben, verwenden Sie in der Anweisung CREATE DATABASE die Parameter USING CODESET und TERRITORY.

**Zugehörige Konzepte:**

- „Unicode-Unterstützung für Systeme zusammengeschlossener Datenbanken“ auf Seite 56
- „Sortierfolgen in einem System zusammengeschlossener Datenbanken“ auf Seite 52

**Zugehörige Tasks:**

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Einstellen der Sortierfolge einer zusammengeschlossenen Datenbank“ auf Seite 55

---

## Sortierfolgen in einem System zusammengeschlossener Datenbanken

Wenn der Server mit zusammengeschlossenen Datenbanken eine Abfrage empfängt, verwendet der SQL-Compiler von DB2<sup>®</sup> Informationen aus dem globalen Katalog und dem Wrapper der Datenquelle, um bei der Verarbeitung der Abfrage zu helfen. Als Teil der Verarbeitung des SQL-Compilers wird die Abfrage vom *Abfrageoptimierungsprogramm* analysiert. Der Compiler entwickelt für die Verarbeitung der Abfrage alternative Strategien, die als *Zugriffspläne* bezeichnet werden. Die Zugriffspläne legen unter Umständen Folgendes fest:

- Die Abfrage soll von den Datenquellen verarbeitet werden.
- Die Abfrage soll vom Server mit zusammengeschlossenen Datenbanken verarbeitet werden.
- Die Abfrage soll zum Teil von den Datenquellen und zum Teil vom Server mit zusammengeschlossenen Datenbanken verarbeitet werden.

Die Durchführung von Zeichensortierungen und -vergleichen in der Datenquelle führt normalerweise zu einer Verbesserung der Leistung.

Sind für die Abfrage Zeichensortierungen oder -vergleiche erforderlich, verwendet der SQL-Compiler Sortierfolgeinformationen zur Ermittlung eines geeigneten Zugriffsplans.

Standardmäßig muss bei der Sortierfolge der zusammengeschlossenen Datenbank die Groß-/Kleinschreibung beachtet werden. Einige Datenquellen des Systems zusammengeschlossener Datenbanken verwenden jedoch Sortierfolgen, die von der Groß-/Kleinschreibung unabhängig sind. Wenn die Sortierfolgen der zusammengeschlossenen Datenbanken und die der Datenquellen voneinander abweichen, kann dies zu unterschiedlichen Abfrageergebnissen führen. Wenn es sich bei der Operation um eine Zeichensortierung handelt, werden zwar dieselben Daten zurückgegeben, doch ist die Reihenfolge der Ergebnisse unterschiedlich. Handelt es sich bei der Operation um einen Zeichenvergleich, sind die zurückgegebenen Ergebnisse möglicherweise unterschiedlich.

Mehrere Faktoren bestimmen, wo die Sortierung oder der Vergleich verarbeitet wird.

- Stimmt die Sortierfolge der zusammengeschlossenen Datenbank mit der Sortierfolge der Datenquelle überein, kann die Zeichensortierung bzw. der Zeichenvergleich auf Datenquellenebene erfolgen. Das Abfrageoptimierungsprogramm kann ermitteln, ob die Abfrage am effizientesten durch eine lokale Operation oder eine ferne Operation verarbeitet wird. Es wird davon ausgegangen, dass alle Typen von Zeichenvergleichen und -sortierungen durch die Datenquelle zu denselben Ergebnissen führen, als würden diese Aktionen von der zusammengeschlossenen Datenbank durchgeführt.
- Unterscheidet sich die Sortierfolge der zusammengeschlossenen Datenbank von der Sortierfolge der Datenquelle und ist die Sortierfolge der Datenquelle von der Groß-/Kleinschreibung abhängig, wird die Zeichensortierung bzw. der Zeichenvergleich auf Ebene der zusammengeschlossenen Datenbank durchgeführt. Es wird davon ausgegangen, dass die Datenquelle dieselben Ergebnisse zu Zeichendaten für die Operationen WHERE=, DISTINCT und GROUP BY ergibt. Andere Operationen wie beispielsweise ORDER BY und WHERE mit einem Vergleichselement 'größer als' oder 'kleiner als' ergeben jedoch unterschiedliche Ergebnisse zu Zeichendaten.
- Unterscheidet sich die Sortierfolge der zusammengeschlossenen Datenbank von der Sortierfolge der Datenquelle und ist die Sortierfolge der Datenquelle von der Groß-/Kleinschreibung unabhängig, wird die Zeichensortierung bzw. der Zeichenvergleich auf Ebene der zusammengeschlossenen Datenbank durchgeführt. Es wird davon ausgegangen, dass die Datenquelle Großbuchstaben und Kleinbuchstaben als äquivalent behandelt und beide in das Ergebnis einschließt, unabhängig davon, ob in der angeforderten Operation Groß- oder Kleinschreibung angegeben wurde. Die Operationen WHERE=, WHERE mit einem Vergleichselement 'größer als' oder 'kleiner als', ORDER BY, DISTINCT und GROUP BY werden nicht im Pushdown-Modus an die Datenquelle gesendet.

Eine von der Groß-/Kleinschreibung unabhängige Datenquelle beispielsweise ordnet den Zeichen "S" und "s" die gleiche Wertigkeit zu. Eine von der Groß-/Kleinschreibung unabhängige Datenquelle mit einer englischen Codepage sieht die Wörter STEWART, SteWArT und stewart als äquivalent an. Wird jedoch eine von der Groß-/Kleinschreibung abhängige Sortierfolge verwendet, werden den Zeichen unterschiedliche Wertigkeiten zugeordnet. In Abhängigkeit von den Angaben zur Groß-/Kleinschreibung für die Sortierfolge können die Ergebnismengen einer Zeichensortierung oder eines Zeichenvergleichs unterschiedlich sein.

Wenn die Sortierfolgen der zusammengeschlossenen Datenbanken und die der Datenquelle voneinander abweichen, ruft der Server mit zusammengeschlossenen Datenbanken die Daten ab und die Sortier- und Vergleichsoperationen werden auf dem lokalen System ausgeführt. Der Grund für die lokale Ausführung dieser Tasks besteht darin, dass die DB2-Benutzer erwarten, dass die Abfrageergebnisse in der Reihenfolge gemäß der für den Server mit zusammengeschlossenen Datenbanken definierten Sortierfolge angezeigt werden. Durch die lokale Sortierung der Daten wird sichergestellt, dass die Ergebnismengen für die DB2-Benutzer konsistent angezeigt werden. Das Abrufen von Daten für lokale Sortier- und Vergleichsoperationen führt normalerweise zu einer Beeinträchtigung der Leistung.

Wenn Sie die Zeichendaten gemäß der Sortierfolge der Datenquelle anzeigen müssen, können Sie Ihre Abfrage in einer Durchgriffssitzung übergeben.

Um zu ermitteln, ob die Sortierfolge der Datenquelle und die Sortierfolge der zusammengesetzten Datenbank einander entsprechen, sind die folgenden Faktoren zu berücksichtigen:

#### **Codepage**

Das Codepageschema (wie beispielsweise ASCII und EBCDIC), das vom Server mit zusammengesetzten Datenbanken und von der Datenquelle verwendet wird, hat einen Einfluss auf die Ergebnisse.

#### **Unterstützung in der Landessprache (NLS)**

Die Sortierfolge steht in Zusammenhang mit der auf einem Server unterstützten Sprache. Vergleichen Sie die DB2-NLS-Informationen des verwendeten Betriebssystems mit den NLS-Informationen der Datenquelle.

#### **Merkmale der Datenquelle**

Einige Datenquellen werden mit Sortierfolgen erstellt, die von der Groß-/Kleinschreibung unabhängig sind. Dies kann bei Operationen, die von einer Reihenfolge abhängig sind, zu unterschiedlichen Ergebnissen aus DB2 führen.

#### **Anpassung**

Einige Datenquellen stellen mehrere Optionen für Sortierfolgen zur Verfügung oder ermöglichen das Anpassen von Sortierfolgen.

Für das Einstellen der Sortierfolge stehen Ihnen folgende Möglichkeiten zur Verfügung:

- Einstellen der Sortierfolge beim Erstellen der zusammengesetzten Datenbank.
- Einstellen der Option `COLLATING_SEQUENCE` beim Erstellen der Serverdefinition für eine Datenquelle. Diese Option ist nur für relationale Datenquellen verfügbar.

#### **Zugehörige Konzepte:**

- „Unicode-Unterstützung für Systeme zusammengesetzter Datenbanken“ auf Seite 56

#### **Zugehörige Tasks:**

- „Erstellen einer zusammengesetzten Datenbank“ auf Seite 58
- „Einstellen der Sortierfolge einer zusammengesetzten Datenbank“ auf Seite 55

#### **Zugehörige Referenzen:**

- „Überlegungen zu Landessprachen in der zusammengesetzten Datenbank“ auf Seite 51

---

## Einstellen der Sortierfolge einer zusammengeschlossenen Datenbank

Administratoren können zusammengeschlossene Datenbanken mit einer bestimmten Sortierfolge erstellen, die mit der Sortierfolge einer Datenquelle übereinstimmt.

Die Einstellung der Sortierfolge der zusammengeschlossenen Datenbank erfolgt im Rahmen der CREATE DATABASE-API. Mit Hilfe dieser API können Sie eine der folgenden Sortierfolgen angeben:

- Eine Identitätssortierfolge
- Eine *Systemsortierfolge* (die Sortierfolge, die von dem Betriebssystem verwendet wird, das die Datenbank unterstützt)
- Eine *angepasste Sortierfolge* (eine vordefinierte Sortierfolge, die von DB2 UDB zur Verfügung gestellt oder von Ihnen selbst definiert wird)

### Vorgehensweise:

Die Sortierfolge der zusammengeschlossenen Datenbank wird mit dem Parameter COLLATE USING in der Anweisung CREATE DATABASE angegeben.

Bei relationalen Datenquellen sollten Sie die Serveroption COLLATING\_SEQUENCE auf 'Y' einstellen, wenn die zusammengeschlossene Datenbank und die Datenquelle die gleiche Sortierfolge verwenden. Wird die Serveroption COLLATING\_SEQUENCE auf 'Y' eingestellt, wird der Server mit zusammengeschlossenen Datenbanken darüber informiert, dass die Sortierfolgen der zusammengeschlossenen Datenbank und der Datenquelle übereinstimmen. Die Einstellung der Serveroption COLLATING\_SEQUENCE erfolgt beim Erstellen der Serverdefinition für die relationalen Datenquellen.

Die Serveroption COLLATING\_SEQUENCE wird von den folgenden relationalen Datenquellen unterstützt:

- DB2-Produktfamilie
- Informix
- Microsoft SQL Server
- ODBC
- OLE DB
- Oracle
- Sybase
- Teradata

### Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammengeschlossener Datenbanken“ auf Seite 56
- „Sortierfolgen in einem System zusammengeschlossener Datenbanken“ auf Seite 52

### Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58

### Zugehörige Referenzen:

- „Überlegungen zu Landessprachen in der zusammengeschlossenen Datenbank“ auf Seite 51

---

## Unicode-Unterstützung für Systeme zusammengesessener Datenbanken

Relationale und nicht relationale Wrapper sowie benutzerdefinierte Funktionen können auf einer Unicode-Datenbank (UTF-8-Datenbank) ausgeführt werden. Die Unicode-Datenbank stellt für Server mit zusammengesessenen Datenbanken plattformunabhängige Umgebungen zur Verfügung. Die Unicode-Datenbank kann Daten bearbeiten, die in verschiedenen Codepages in unterschiedlichen Datenquellen gespeichert wurden.

Folgende Wrapper und benutzerdefinierte Funktionen unterstützen Unicode:

- Relationale Wrapper
  - DRDA<sup>®</sup>
  - Informix<sup>®</sup>
  - MS SQL Server
  - ODBC
  - OLE DB
  - Oracle
  - Sybase
  - Teradata
- Nicht relationale Wrapper und benutzerdefinierte Funktionen
  - BioRS-Wrapper
  - BLAST-Wrapper
  - Documentum-Wrapper
  - Entrez-Wrapper
  - Excel-Wrapper
  - HMMER-Wrapper
  - IBM<sup>®</sup> Lotus<sup>®</sup> Extended Search-Wrapper
  - Benutzerdefinierte KEGG-Funktionen
  - Benutzerdefinierte MQ-Funktionen
  - Wrapper für Dateien mit Tabellenstruktur
  - Benutzerdefinierte Web-Services-Funktionen
  - Web-Services-Wrapper
  - WebSphere<sup>®</sup> Business Integration-Wrapper
  - XML-Wrapper

In Abb. 5 auf Seite 57 sehen Sie das Beispiel eines Unternehmens mit Niederlassungen in verschiedenen Ländern. Jede Niederlassung speichert Kundendaten in ihrer eigenen Datenbank und mit einer eigenen Codepage. Die Datenbank von Microsoft<sup>®</sup> SQL Server speichert Daten in Codepage A. Die Oracle-Datenbank speichert Daten in Codepage B. Codepage A gilt für ein anderes Gebiet als Codepage B. Um die Daten der verschiedenen Gebiete zu integrieren, kann das Unternehmen als Codepage der zusammengesessenen Datenbank Unicode verwenden. Anschließend kann das Unternehmen die Tabellen verknüpfen, um die Gesamtzahl aller Bestellungen unabhängig vom jeweiligen Gebiet anzuzeigen.

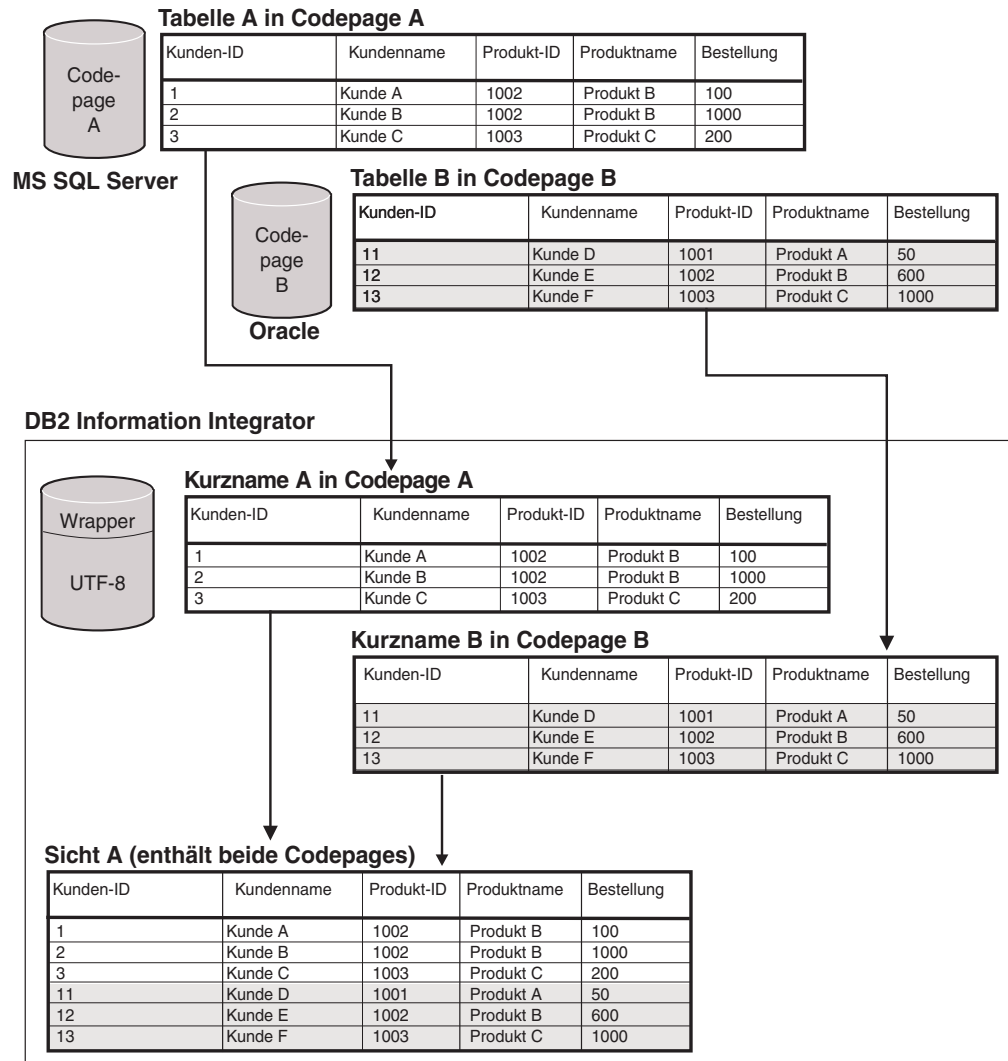


Abbildung 5. Unicode - Beispiel

#### Zugehörige Tasks:

- „Angabe der Client-Codepage für die Unicode-Unterstützung von Microsoft SQL Server- und ODBC-Datenquellen“ in *Systeme zusammenschlossener Datenbanken*
- „Angabe der Dateicodepage für die Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur“ in *Systeme zusammenschlossener Datenbanken*

#### Zugehörige Referenzen:

- „Standardmäßige progressive Typzuordnungen in Unicode - NET8-Wrapper“ in *Systeme zusammenschlossener Datenbanken*
- „Unterstützte Unicode-Codepages für die CODEPAGE-Option des MSSQL- und ODBC-Wrappers“ in *Systeme zusammenschlossener Datenbanken*
- „Standardmäßige degressive Typzuordnungen in Unicode - NET8-Wrapper“ in *Systeme zusammenschlossener Datenbanken*
- „Standardmäßige progressive Typzuordnungen in Unicode - Sybase-Wrapper“ in *Systeme zusammenschlossener Datenbanken*
- „Standardmäßige degressive Typzuordnungen in Unicode - Sybase-Wrapper“ in *Systeme zusammenschlossener Datenbanken*

- „Standardmäßige progressive Typzuordnungen in Unicode - ODBC-Wrapper“ in *Systeme zusammengeschlossener Datenbanken*
- „Standardmäßige degressive Typzuordnungen in Unicode - ODBC-Wrapper“ in *Systeme zusammengeschlossener Datenbanken*
- „Standardmäßige progressive Typzuordnungen in Unicode - Microsoft SQL Server-Wrapper“ in *Systeme zusammengeschlossener Datenbanken*
- „Standardmäßige degressive Typzuordnungen in Unicode - Microsoft SQL Server-Wrapper“ in *Systeme zusammengeschlossener Datenbanken*
- „Angeben der Dateicodpage für die Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur - Beispiel“ in *Systeme zusammengeschlossener Datenbanken*

---

## Erstellen einer zusammengeschlossenen Datenbank

Nach der Konfiguration des Servers mit zusammengeschlossenen Datenbanken muss der DB2-Exemplareigner eine DB2-Datenbank auf dem Exemplar des Servers mit zusammengeschlossenen Datenbanken erstellen, das als zusammengeschlossene Datenbank fungieren soll.

**Empfehlung:** Wenn die fernen Datenquellen, zu denen Sie eine Verbindung herstellen müssen, unterschiedliche oder nicht kompatible Codepages verwenden, definieren Sie die zusammengeschlossene Datenbank als Unicode-Datenbank. Um die zusammengeschlossene Datenbank als Unicode-Datenbank zu definieren, geben Sie USING CODESET UTF-8 in der Anweisung CREATE DATABASE an.

Dieser Schritt muss ausgeführt werden, bevor Sie den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Ihre Datenquellen konfigurieren können.

### Voraussetzungen:

- Sie benötigen die Berechtigung SYSADM oder SYSCTRL zum Erstellen einer DB2-Datenbank.
- DB2 Information Integrator muss auf einem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Überlegen Sie, ob Sie beim Erstellen der zusammengeschlossenen Datenbank eine Sortierfolge angeben wollen.

### Vorgehensweise:

Sie können die zusammengeschlossene Datenbank über die DB2-Steuerzentrale oder die DB2-Befehlszeile erstellen.

Um diese Task über die DB2-Steuerzentrale auszuführen, gehen Sie wie folgt vor:

1. Klicken Sie zunächst den Ordner **Datenbanken** mit Maustaste 2 an und klicken Sie anschließend **Erstellen ->Datenbank mit Assistent** an. Das Fenster 'Assistent: Datenbank erstellen' wird geöffnet.
2. Führen Sie die Schritte im Assistenten aus.

| Um diese Task über die DB2-Befehlszeile auszuführen, setzen Sie den Befehl CRE-  
| ATE DATABASE ab. Beispiel:

| CREATE DATABASE *zusammengeschlossene\_datenbank*

| Mit diesem Befehl werden folgende Aktionen ausgeführt:

- | • Initialisierung einer neuen Datenbank
- | • Erstellen der drei anfänglichen Tabellenbereiche
- | • Erstellen der Systemtabellen
- | • Zuordnen des Wiederherstellungsprotokolls

| Wenn Ihr DB2-Exemplar eine Konfiguration mit mehreren Partitionen verwendet,  
| wirkt sich der Befehl CREATE DATABASE auf alle Partitionen aus, die in der Datei  
| *db2nodes.cfg* aufgeführt sind. Die Datenbankpartition, über die dieser Befehl abge-  
| setzt wird, fungiert als Katalogpartition für die neue Datenbank.

| **Zugehörige Konzepte:**

- | • „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2*  
| *Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- | • „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

| **Zugehörige Tasks:**

- | • „Einstellen der Sortierfolge einer zusammengesetzten Datenbank“ auf Seite  
| 55





---

## Teil 3. Datenquellen



---

## Kapitel 5. Konfigurieren des Zugriffs auf Datenquellen - Überblick

In den folgenden Abschnitten wird genau erklärt, wie Sie einen Server mit zusammengeschlossenen Datenbanken und eine zusammengeschlossene Datenbank für den Zugriff auf Ihre Datenquellen konfigurieren:

- Sie enthalten Informationen über die erforderlichen Grundschrirte für eine schnelle Konfiguration.
- Sie umreißen verschiedene optionale Schritte, die eventuell für die Feinabstimmung der Datenquellenkonfiguration erforderlich sind.

Für jede Datenquelle gibt es ein eigenes Kapitel zur Konfiguration.

---

### Konfigurieren von Datenquellen - Kurzüberblick

Die erforderlichen Schritte zum Konfigurieren des Zugriffs auf eine Datenquelle können größtenteils mit der DB2®-Steuerzentrale ausgeführt werden. Verwenden Sie für die Schritte, die eine Befehlszeile erfordern, die DB2-Befehlszentrale. Schalten Sie zwischen diesen graphischen Benutzerschnittstellen hin und her, um den Zugriff auf eine Datenquelle schnell zu konfigurieren.

Bevor Sie den Zugriff auf eine Datenquelle konfigurieren, stellen Sie sicher, dass der Server mit zusammengeschlossenen Datenbanken korrekt eingerichtet wurde.

Die Schritte zur Konfiguration des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen sind für jede Datenquelle ähnlich. Nachfolgend sind die Grundschrirte sowie die empfohlenen Schnittstellen aufgeführt:

*Tabelle 11. Empfohlene Schnittstelle und Konfigurationsschrirte*

Konfigurationsschrirte	Empfohlene Schnittstelle	Anmerkungen
1. Server mit zusammengeschlossenen Datenbanken für die Datenquelle vorbereiten.	Clientkonfiguration - Unterstützung	Nur für einige Datenquellen erforderlich. Für die Ausführung dieses Schrittes ist unter Umständen die Installation von Software, das Konfigurieren einer Datei oder die Prüfung einer Einstellung erforderlich.
2. Erforderliche Umgebungsvariablen festlegen.	DB2-Steuerzentrale	Umgebungsvariablen sind erforderlich für: <ul style="list-style-type: none"><li>• Documentum</li><li>• Informix®</li><li>• Microsoft® SQL Server</li><li>• Oracle</li><li>• Sybase</li><li>• Teradata</li></ul>
2. Wrapper registrieren.	Assistent für zusammengeschlossene Objekte in der DB2-Steuerzentrale.	Für jede Datenquelle, auf die Sie zugreifen möchten, ist ein Wrapper erforderlich.

Tabelle 11. Empfohlene Schnittstelle und Konfigurationsschritte (Forts.)

Konfigurationsschritt	Empfohlene Schnittstelle	Anmerkungen
3. Serverdefinitionen registrieren.	Assistent für zusammengeslossene Objekte in der DB2-Steuerzentrale.	Die Serverdefinitionen werden einem Wrapper zugeordnet und beim Registrieren von Kurznamen verwendet.
4. Benutzerzuordnungen erstellen.	Assistent für zusammengeslossene Objekte in der DB2-Steuerzentrale.	Nur für einige Datenquellen erforderlich.  Wenn Sie versuchen, das einer Benutzerzuordnung aus der Katalogsicht SYSCAT.USEROPTIONS zugeordnete ferne Kennwort abzurufen, wird der Wert für das ferne Kennwort verschlüsselt angezeigt.
5. Verbindung zum Datenquellenserver testen.	DB2-Befehlszentrale	Nur für einige Datenquellen erforderlich.
6. Kurznamen erstellen.	Assistent für zusammengeslossene Objekte in der DB2-Steuerzentrale.	Für jedes Datenquellenobjekt, auf das Sie zugreifen möchten, ist ein Kurzname erforderlich.

#### Zugehörige Konzepte:

- „Optionale Konfigurationsschritte“ auf Seite 76

#### Zugehörige Tasks:

- „Hinzufügen von Datenquellen zu einem Server mit zusammengesetzten Datenbanken mit Hilfe der DB2 UDB-Steuerzentrale“ auf Seite 64
- „Registrieren von Wrappern für eine Datenquelle“ auf Seite 70
- „Registrieren von Serverdefinitionen für eine Datenquelle“ auf Seite 71
- „Registrieren von Benutzerzuordnungen für eine Datenquelle“ auf Seite 72
- „Registrieren von Kurznamen für Datenquellen“ auf Seite 73
- „Festlegen der Umgebungsvariablen für eine Datenquelle“ auf Seite 67
- „Überprüfung der Konfiguration des Servers mit zusammengesetzten Datenbanken“ auf Seite 43

## Hinzufügen von Datenquellen zu einem Server mit zusammengesetzten Datenbanken mit Hilfe der DB2 UDB-Steuerzentrale

Um den Server mit zusammengesetzten Datenbanken für den Zugriff auf Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

#### Vorgehensweise:

Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengesetzten Datenbanken führt. Um den Assistenten zu starten, klicken Sie mit der rechten Maustaste den Ordner **Objekte zusammengesetzter Datenbanken** an, und klicken Sie anschließend **Zusammengesetzte Objekte erstellen** an.

Die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken sind für jede Datenquelle unterschiedlich.

Sie können mehrere Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenbanken konfigurieren, indem Sie das Fenster **Aktionsausgabe** verwenden.

#### **Zugehörige Tasks:**

- „Konfigurieren von mehreren Servern mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen“ auf Seite 66
- „Hinzufügen von Datenquellen von Dateien mit Tabellenstruktur zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 406
- „Hinzufügen von Documentum-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 193
- „Hinzufügen von Excel-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 250
- „Hinzufügen von BLAST-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 113
- „Hinzufügen von Entrez-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 222
- „Hinzufügen von Teradata-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 415
- „Hinzufügen von BioRS-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 79
- „Hinzufügen von Datenquellen der DB2-Produktfamilie zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 179
- „Hinzufügen von Extended Search-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 270
- „Hinzufügen von HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 290
- „Hinzufügen von Informix-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken“ auf Seite 313
- „Hinzufügen von Microsoft SQL Server-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 331
- „Hinzufügen von ODBC-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 347
- „Hinzufügen von OLE DB-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 365
- „Hinzufügen von Oracle-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 371
- „Hinzufügen von Sybase-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 387
- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengeschlossener Datenbanken“ auf Seite 144
- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 439

---

## Konfigurieren von mehreren Servern mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen

Ein System zusammengeschlossener Datenbanken kann aus mehreren Servern mit zusammengeschlossenen Datenbanken bestehen. Anstatt jeden Server mit zusammengeschlossenen Datenbanken separat zu konfigurieren, können Sie mit Hilfe der DB2-Steuerzentrale die Konfiguration der Server zeitsparend durchführen. Beim Konfigurieren des ersten Servers erfasst das Fenster **Aktionsausgabe** die DDL-Anweisungen, die beim Erstellen der zusammengeschlossenen Objekte ausgegeben werden. Sie können diese Anweisungen erneut verwenden oder modifizieren. Sie werden angewendet, um zusätzliche Server mit zusammengeschlossenen Datenbanken schnell zu konfigurieren.

Das Fenster **Aktionsausgabe** bleibt für die Dauer der aktuellen Sitzung aktiv. Wenn Sie das Fenster **Aktionsausgabe** schließen, werden die DDL-Anweisungen für die aktuelle Sitzung weiter im Fenster **Aktionsausgabe** gespeichert. Wenn Sie jedoch die DB2-Steuerzentrale schließen, werden alle DDL-Anweisungen aus der aktuellen Sitzung aus dem Fenster **Aktionsausgabe** entfernt.

### Voraussetzungen:

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.

### Vorgehensweise:

Gehen Sie wie folgt vor, um mehrere Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen zu konfigurieren:

1. Verwenden Sie die DB2-Steuerzentrale, um den ersten Server mit zusammengeschlossenen Datenbanken für die Datenquellen zu konfigurieren, auf die Sie zugreifen möchten. Damit wird jede DDL-Anweisung erfasst.
2. Zeigen Sie die Seite **Aktionsausgabe** im Fenster **Aktionsausgabe** an.  
Wenn Sie das Fenster **Aktionsausgabe** geschlossen haben, klicken Sie mit der rechten Maustaste den Ordner **Objekte zusammengeschlossener Datenbanken** an, und klicken Sie anschließend **Aktionen anzeigen** an, um das Fenster **Aktionsausgabe** zu öffnen.
3. Löschen Sie diejenigen DDL-Anweisungen, die Sie nicht für andere Server mit zusammengeschlossenen Datenbanken verwenden möchten. Um eine Anweisung zu löschen, klicken Sie die Anweisung mit der rechten Maustaste an, und klicken Sie anschließend **Entfernen** an. Möglicherweise möchten Sie z. B. alle Anweisungen löschen, die auf der Seite **Aktionsausgabe** in der Statusspalte den Status 'Fehlgeschlagen' anzeigen.
4. Kopieren Sie die Anweisungen, die Sie für die anderen Server mit zusammengeschlossenen Datenbanken verwenden möchten, in die Seite **Befehlseditor**:
  - a. Wählen Sie die Anweisungen aus, die Sie kopieren möchten. Um mehrere Anweisungen auszuwählen, verwenden Sie die Strg-Taste.
  - b. Klicken Sie mit der rechten Maustaste die ausgewählten Anweisungen an, und klicken Sie anschließend **In Befehlseditor kopieren** an. Die Seite **Befehlseditor** wird geöffnet.

5. Ändern Sie die DDL-Anweisungen, die Sie für die anderen Server mit zusammengeschlossenen Datenbanken verwenden möchten, in der Seite **Befehlseditor**. Möglicherweise möchten Sie z. B. alle Anweisungen ändern, die ein lokales Schema angeben.

Um die Kennwörter anzugeben, müssen Sie die Benutzerzuordnungen ändern. Ist die DDL für die Anweisungen CREATE USER MAPPING im Fenster **Aktionsausgabe** erfasst, erscheinen die Kennwörter mit Sternchen als Platzhalterzeichen. Sie müssen die Sternchen durch richtige Kennwörter ersetzen.

6. Geben Sie die DDL-Anweisungen für den nächsten Server mit zusammengeschlossenen Datenbanken aus.

#### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

#### Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58

---

## Einstellen der Umgebungsvariablen für Datenquellen

### Festlegen der Umgebungsvariablen für eine Datenquelle

Wenn Sie DB2 Information Integrator installieren, versucht der Installationsprozess, die Umgebungsvariablen zu setzen, die für einige Datenquellen erforderlich sind. Der Installationsprozess kann unter Umständen die Umgebungsvariablen nicht setzen, wenn z. B. vor der Installation von DB2 Information Integrator nicht die Client-Software auf dem Server mit zusammengeschlossenen Datenbanken installiert wurde.

Wenn Sie den Schritten zum Hinzufügen einer Datenquelle zum Server mit zusammengeschlossenen Datenbanken folgen, können Sie die Umgebungsvariablen prüfen und (falls erforderlich) setzen.

- Wenn Sie die DB2-Steuerzentrale verwenden, um Datenquellen zum Server mit zusammengeschlossenen Datenbanken hinzuzufügen, werden die Anforderungen für die Umgebungsvariablen automatisch geprüft. Sie können die Umgebungsvariablen beim Erstellen oder Ändern eines Wrappers setzen.
- Wenn Sie die DB2-Befehlszeile zum Hinzufügen von Datenquellen zum Server mit zusammengeschlossenen Datenbanken verwenden, müssen Sie die Umgebungsvariablen manuell setzen.

Das Setzen der Umgebungsvariablen ist für folgende Datenquellen erforderlich:

- Documentum
- Informix
- Microsoft SQL Server
- Oracle
- Sybase
- Teradata



### **Voraussetzungen:**

Diese Task sollte vom Systemadministrator ausgeführt werden.

### **Einschränkungen:**

Siehe hierzu den Abschnitt: Einschränkungen für die Datei db2dj.ini

### **Vorgehensweise:**

Die auszuführenden Schritte zum Prüfen der Umgebungsvariablen sind je nach Datenquelle unterschiedlich.

### **Zugehörige Konzepte:**

- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

### **Zugehörige Tasks:**

- „Festlegen der Documentum-Umgebungsvariablen“ auf Seite 195
- „Registrieren von Wrappern für eine Datenquelle“ auf Seite 70
- „Festlegen der Informix-Umgebungsvariablen“ auf Seite 315
- „Festlegen der Umgebungsvariablen für Microsoft SQL Server“ auf Seite 334
- „Einrichten der Oracle-Umgebungsvariablen“ auf Seite 372
- „Einstellen der Sybase-Umgebungsvariablen“ auf Seite 388
- „Einstellen der Teradata-Umgebungsvariablen“ auf Seite 418

### **Zugehörige Referenzen:**

- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68

## **Einschränkungen für die Datei 'db2dj.ini'**

Für die Datei db2dj.ini gelten die folgenden Einschränkungen:

- Alle Eingaben müssen in folgendem Format erfolgen: *umgebungsvariable=wert*. Hierbei ist *umgebungsvariable* der Name der Umgebungsvariable und *wert* ist der ihr zugeordnete Wert.
- Der Name der Umgebungsvariablen darf maximal 255 Byte lang sein.
- Der Wert der Umgebungsvariable darf maximal 765 Byte lang sein.
- Jede Zeile in der Datei darf maximal 1021 Byte lang sein. Alle Daten, die über diese Länge hinausgehen, werden ignoriert.

### **Zugehörige Tasks:**

- „Festlegen der Documentum-Umgebungsvariablen“ auf Seite 195
- „Festlegen der Umgebungsvariablen für eine Datenquelle“ auf Seite 67
- „Festlegen der Informix-Umgebungsvariablen“ auf Seite 315
- „Festlegen der Umgebungsvariablen für Microsoft SQL Server“ auf Seite 334
- „Einrichten der Oracle-Umgebungsvariablen“ auf Seite 372
- „Einstellen der Sybase-Umgebungsvariablen“ auf Seite 388
- „Einstellen der Teradata-Umgebungsvariablen“ auf Seite 418

## Anwenden von Umgebungsvariablen in einer Exemplarkonfiguration mit mehreren Partitionen

Wenn Ihr Exemplar des Servers mit zusammengeschlossenen Datenbanken über eine Konfiguration mit mehreren Partitionen verfügt, müssen Sie die Umgebungsvariablen für Datenquellen auf alle Partitionen anwenden.

Dieser Schritt ist nur dann erforderlich, wenn der Server mit zusammengeschlossenen Datenbanken eine Exemplarkonfiguration mit mehreren Partitionen hat.

Die Umgebungsvariablen für Datenquellen sind in der Datei `db2dj.ini` enthalten. Diese Datei wurde dem Server mit zusammengeschlossenen Datenbanken bei der Installation von DB2 Information Integrator hinzugefügt.

Sie müssen jeweils eine Kopie derselben Datei `db2dj.ini` allen Partitionen in Ihrer Exemplarkonfiguration hinzufügen. Der Standardname der Datei lautet `db2dj.ini`. Auf UNIX-Servern mit zusammengeschlossenen Datenbanken ist `INSTHOME/sqllib/cfg` der Standardpfad für die Datei `db2dj.ini`, wobei `INSTHOME` das Ausgangsverzeichnis des Exempleigners ist. Auf Windows-Servern mit zusammengeschlossenen Datenbanken lautet der Standardpfad zu dieser Datei `x:\SQLLIB\cfg`, wobei `x:\SQLLIB` das Laufwerk und das Verzeichnis ist, die in der Registrierdatenbankvariablen bzw. Umgebungsvariablen `DB2PATH` angegeben wurden.

Standardpfad und Standardname der Datei `db2dj.ini` können mit Hilfe der Registrierdatenbankvariablen `DB2_DJ_INI` überschrieben werden.

Sie können die Registrierdatenbankvariable `DB2_DJ_INI` entweder auf alle Partitionen oder auf eine Untergruppe der Partitionen anwenden.

### Vorgehensweise:

Um die Registrierdatenbankvariable `DB2_DJ_INI` auf die entsprechenden Partitionen auf Ihrem Server mit zusammengeschlossenen Datenbanken anzuwenden, setzen Sie den Befehl `db2set` ab. Mit dem Befehl `db2set` können DB2-Profilvariablen angezeigt, eingestellt oder entfernt werden.

Die Syntax des zu verwendenden Befehls `db2set` hängt von der jeweiligen Struktur des Datenbanksystems ab.

- Um die Registrierdatenbankvariable `DB2_DJ_INI` auf alle Datenbankpartitionen innerhalb des betreffenden Exemplars anzuwenden, setzen Sie den folgenden Befehl ab:

```
db2set -g DB2_DJ_INI=$HOME/sqllib/cfg/my_db2dj.ini
```

- Um die Registrierdatenbankvariable `DB2_DJ_INI` lediglich auf die aktuelle Partition anzuwenden, setzen Sie den folgenden Befehl ab:

```
db2set DB2_DJ_INI=$HOME/sqllib/cfg/my_db2dj.ini
```

- Um die Registrierdatenbankvariable `DB2_DJ_INI` auf eine bestimmte Partition anzuwenden, setzen Sie den folgenden Befehl ab:

```
db2set -i EXEMPLARX 3 DB2_DJ_INI=$HOME/sqllib/cfg/partition3.ini
```

*EXEMPLARX*

Der Name des Exemplars.

3 Die Partitionsnummer gemäß Liste in der Datei `db2nodes.cfg`.

*partition3.ini*

Die geänderte und umbenannte Version der Datei db2dj.ini.

**Achtung:** wird die Registrierdatenbankvariable DB2\_DJ\_INI eingestellt, muss sie auf einen absoluten Pfad gesetzt werden. Wenn der Parameter FEDERATED auf YES (Ja) eingestellt ist und die Registrierdatenbankvariable DB2\_DJ\_INI auf einen relativen Pfad gesetzt wird, kann die DB2 Universal Database-Steuerkomponente nicht gestartet werden.

**Zugehörige Tasks:**

- „Festlegen der Umgebungsvariablen für eine Datenquelle“ auf Seite 67

---

## Registrieren von Wrappern für eine Datenquelle

Sie registrieren einen Wrapper für jeden Typ von Datenquelle, auf den Sie zugreifen möchten. Um auf drei DB2 für z/OS-Datenbanktabellen, eine DB2 für iSeries-Tabelle und zwei BLAST-Suchtypen zuzugreifen, müssen Sie zwei Wrapper erstellen: einen für die DB2-Datenbanken und einen für die BLAST-Suchtypen.

Nachdem diese Wrapper in der zusammengeschlossenen Datenbank registriert sind, können sie für den Zugriff auf andere Objekte in diesen Datenquellen verwendet werden. Sie können beispielsweise den DRDA-Wrapper für den Zugriff auf Datenquellen von allen Datenquellenobjekten der DB2-Produktfamilie verwenden, einschließlich DB2 für Linux, UNIX und Windows, DB2 für z/OS und OS/390, DB2 für iSeries und DB2 Server für VM und VSE.

**Vorgehensweise:**

Sie können einen Wrapper von der DB2-Steuerzentrale aus oder über die DB2-Befehlszeile registrieren:

- Um diese Task von der DB2-Steuerzentrale aus auszuführen, verwenden Sie den Assistenten für zusammengeschlossene Objekte oder klicken Sie mit der rechten Maustaste den Ordner **Zusammengeschlossene Objekte** an, und klicken Sie anschließend **Wrapper erstellen** an.
- Um diese Task von der DB2-Befehlszeile aus auszuführen, verwenden Sie die Anweisung CREATE WRAPPER.

**Zugehörige Tasks:**

- „Registrieren des Wrappers für Dateien mit Tabellenstruktur“ auf Seite 407
- „Registrieren des Documentum-Wrappers“ auf Seite 196
- „Registrieren des Excel-Wrappers“ auf Seite 251
- „Registrieren des BLAST-Wrappers“ auf Seite 121
- „Registrieren des XML-Wrappers“ auf Seite 475
- „Registrieren des Entrez-Wrappers“ auf Seite 225
- „Registrieren des Teradata-Wrappers“ auf Seite 421
- „Registrieren des BioRS-Wrappers“ auf Seite 81
- „Registrieren des DB2-Wrappers“ auf Seite 182
- „Registrieren des Extended Search-Wrappers“ auf Seite 271
- „Registrieren des HMMER-Wrappers“ auf Seite 299
- „Registrieren des Informix-Wrappers“ auf Seite 319
- „Registrieren des Microsoft SQL Server-Wrappers“ auf Seite 336

- „Registrieren des ODBC-Wrappers“ auf Seite 350
- „Registrieren des OLE DB-Wrappers“ auf Seite 366
- „Registrieren des Oracle-Wrappers“ auf Seite 377
- „Registrieren des Sybase-Wrappers“ auf Seite 392
- „Registrieren des WebSphere Business Integration-Wrappers“ auf Seite 145
- „Registrieren des Web-Services-Wrappers“ auf Seite 440

---

## Registrieren von Serverdefinitionen für eine Datenquelle

Der Zweck einer Serverdefinition ist je nach Datenquelle unterschiedlich.

Eine Serverdefinition für relationale Datenquellen stellt gewöhnlich eine ferne Datenbank, eine Datenbankpartition oder einen Knoten dar. Bei nicht relationalen Datenquellen werden manche Serverdefinitionen zu einem Suchtyp und Dämon, zu einer Website oder zu einem Web-Server zugeordnet. Bei anderen nicht relationalen Datenquellen wird die Serverdefinition nur erstellt, da dies durch den Zusammenschluss von Datenquellen erfordert wird.

Jedes Datenquellenobjekt, für das Sie einen Kurznamen erstellen, muss einer bestimmten Serverdefinition zugeordnet werden.

Bei einigen Datenquellen müssen Sie einen Knoten angeben, wenn Sie eine Serverdefinition registrieren. Das Konzept eines Knotens ist je nach Datenquelle unterschiedlich. Bei relationalen Datenquellen reflektiert ein Knoten ein Serverexemplar der Datenquelle.

### Vorgehensweise:

Sie können eine Serverdefinition von der DB2-Steuerzentrale aus oder über die DB2-Befehlszeile registrieren:

- Um diese Task von der DB2-Steuerzentrale aus auszuführen, verwenden Sie den Assistenten für zusammengesessene Objekte oder klicken Sie mit der rechten Maustaste den Ordner **Serverdefinitionen** an, und klicken Sie anschließend **Erstellen** an.
- Um diese Task von der DB2-Befehlszeile aus auszuführen, verwenden Sie die Anweisung CREATE SERVER.

### Zugehörige Tasks:

- „Registrieren des Servers für Dateien mit Tabellenstruktur“ auf Seite 408
- „Registrieren des Servers für Documentum-Datenquellen“ auf Seite 198
- „Registrieren des Servers für eine Excel-Datenquelle“ auf Seite 253
- „Registrieren des Servers für BLAST-Datenquellen“ auf Seite 122
- „Registrieren des Servers für XML-Datenquellen“ auf Seite 477
- „Registrieren des Servers für eine Entrez-Datenquelle“ auf Seite 226
- „Registrieren der Serverdefinitionen für eine Teradata-Datenquelle“ auf Seite 422
- „Registrieren der Serverdefinition für eine BioRS-Datenquelle“ auf Seite 83
- „Registrieren der Serverdefinitionen für eine DB2-Datenquelle“ auf Seite 184
- „Registrieren des Servers für Extended Search-Datenquellen“ auf Seite 272
- „Registrieren der Serverdefinition für eine HMMER-Datenquelle“ auf Seite 300
- „Registrieren der Serverdefinitionen für eine Informix-Datenquelle“ auf Seite 321

- „Registrieren der Serverdefinitionen für eine Microsoft SQL Server-Datenquelle“ auf Seite 338
- „Registrieren der Serverdefinitionen für eine ODBC-Datenquelle“ auf Seite 352
- „Registrieren der Serverdefinitionen für eine OLE DB-Datenquelle“ auf Seite 367
- „Registrieren der Serverdefinitionen für eine Oracle-Datenquelle“ auf Seite 378
- „Registrieren der Serverdefinitionen für eine Sybase-Datenquelle“ auf Seite 394
- „Registrieren der Serverdefinition für Geschäftsanwendungsdatenquellen“ auf Seite 146
- „Registrieren der Serverdefinition für Web-Services-Datenquellen“ auf Seite 441

---

## Registrieren von Benutzerzuordnungen für eine Datenquelle

Bei einigen Datenquellen müssen Sie eine Zuordnung zwischen der Berechtigungs-ID des Servers mit zusammengeschlossenen Datenbanken und der Benutzer-ID mit Kennwort für die Datenquelle definieren. Diese Zuordnung muss für alle Benutzer-IDs erstellt werden, die das System zusammengeschlossener Datenbanken zum Senden verteilter Anforderungen nutzen.

### Vorgehensweise:

Sie können eine Benutzerzuordnung von der DB2-Steuerzentrale aus oder über die DB2-Befehlszeile erstellen:

- Um diese Task von der DB2-Steuerzentrale aus auszuführen, verwenden Sie den Assistenten für zusammengeschlossene Objekte oder klicken Sie mit der rechten Maustaste den Ordner **Benutzerzuordnungen** an, und klicken Sie anschließend **Erstellen** an.
- Um diese Task von der DB2-Befehlszeile aus auszuführen, verwenden Sie die Anweisung CREATE USER MAPPING.

### Zugehörige Tasks:

- „Registrieren der Benutzerzuordnungen für Documentum-Datenquellen“ auf Seite 198
- „Erstellen der Benutzerzuordnung für eine Teradata-Datenquelle“ auf Seite 425
- „Registrieren von Benutzerzuordnungen für BioRS-Datenquellen“ auf Seite 83
- „Erstellen der Benutzerzuordnung für eine DB2-Datenquelle“ auf Seite 186
- „Registrieren von Benutzerzuordnungen für Extended Search-Datenquellen“ auf Seite 273
- „Erstellen der Benutzerzuordnung für eine Informix-Datenquelle“ auf Seite 323
- „Erstellen einer Benutzerzuordnung für eine Microsoft SQL Server-Datenquelle“ auf Seite 341
- „Erstellen einer Benutzerzuordnung für eine ODBC-Datenquelle“ auf Seite 354
- „Erstellen einer Benutzerzuordnung für eine OLE DB-Datenquelle“ auf Seite 369
- „Erstellen der Benutzerzuordnungen für eine Oracle-Datenquelle“ auf Seite 381
- „Erstellen einer Benutzerzuordnung für eine Sybase-Datenquelle“ auf Seite 397

---

## Registrieren von Kurznamen für Datenquellen

### Registrieren von Kurznamen für Datenquellen

Das Registrieren eines Kurznamens ist typischerweise die umfangreichste Konfigurationstask. Die auszuführenden Schritte und Anforderungen für das Registrieren eines Kurznamens sind je nach Datenquelle unterschiedlich.

**Empfehlung:** Verwenden Sie die DB2-Steuerzentrale zum Registrieren von Kurznamen. Für die meisten Datenquellen kann das Tool zum Aufspüren (Discover-Tool) in der DB2-Steuerzentrale verwendet werden. Mit Hilfe des Discover-Tools können Sie Datenquellenobjekte schnell identifizieren, für die Sie möglicherweise Kurznamen registrieren möchten.

Für jedes Datenquellenobjekt, auf das Sie zugreifen möchten, muss ein Kurzname erstellt werden.

Datenquellenobjekte können relational oder nicht relational sein:

- Beispiele für relationale Datenquellenobjekte sind Datenbanktabellen, Sichten und Synonyme (nur Informix und Oracle)
- Beispiele für nicht relationale Datenquellenobjekte sind BLAST-fähige Datenbanken, Objekte und registrierte Tabellen in einer Documentum Docbase, Microsoft Excel-Dateien (.xls), Dateien mit Tabellenstruktur (.txt) und XML-Dateien.

Tabellen und Sichten, die sich in der zusammengeschlossenen Datenbank befinden, sind *lokale Objekte*. Für diese Objekte werden keine Kurznamen registriert. In den Abfragen wird der tatsächliche Objektname verwendet.

*Datenquellenobjekte* sind:

- Tabellen und Sichten eines anderen DB2®-Datenbankexemplars auf dem Server mit zusammengeschlossenen Datenbanken.
- Tabellen und Sichten in einem DB2-Exemplar auf einem anderen Server.
- Datenquellenobjekte, die sich in einer anderen Datenquelle, wie z. B. Oracle, Sybase, Documentum oder ODBC, befinden.

Für diese Objekte müssen Kurznamen erstellt werden. Datenquellenobjekte werden manchmal als *ferne Objekte* bezeichnet.

Wenn Sie eine verteilte Anforderung an den Server mit zusammengeschlossenen Datenbanken übergeben, verweist die Anforderung unter Verwendung des Kurznamens auf ein Datenquellenobjekt. Kurznamen werden bestimmten Objektnamen in der Datenquelle zugeordnet. Durch die Zuordnungen ist keine Qualifikation der Kurznamen durch Datenquellennamen erforderlich. Die Speicherposition der Datenquellenobjekte ist für die Clientanwendung oder den Endbenutzer transparent. Bei Kurznamen handelt es sich nicht um alternative Namen für Datenquellenobjekte. Sie sind Zeiger, mit denen der Server mit zusammengeschlossenen Datenbanken auf diese Objekte verweist.

Wenn Sie beispielsweise den Kurznamen *DEPT* zur Darstellung einer Informix-Datenbank namens *NFX1.PERSON.DEPT* definieren, wird die Anweisung `SELECT * FROM DEPT` vom Server mit zusammengeschlossenen Datenbanken akzeptiert. Die Anweisung `SELECT * FROM NFX1.PERSON.DEPT` ist dagegen nicht erlaubt.



Wenn Sie einen Kurznamen registrieren, werden Metadateninformationen zu diesem Kurznamen im Katalog des Systems zusammengeschlossener Datenbanken gespeichert. Für ein Objekt einer relationalen Datenquelle werden die Katalogdaten vom fernen Server abgerufen und im Katalog des Systems zusammengeschlossener Datenbanken gespeichert. Bei nicht relationalen Datenquellen hängt die Art der Speicherung von Datenquelleninformationen im Katalog des Systems zusammengeschlossener Datenbanken von der jeweiligen Datenquelle ab. Die Informationen werden möglicherweise vom fernen Server abgerufen oder sie müssen in die Anweisung CREATE NICKNAME eingebunden werden.

Der SQL-Compiler verwendet diese Metadaten, um den Zugriff auf das Datenquellenobjekt zu erleichtern. Wenn beispielsweise ein Kurzname für eine Tabelle mit einem Index registriert wird, werden die auf den Index bezogenen Metadaten im Katalog des Systems zusammengeschlossener Datenbanken gespeichert. Wenn Sie den Kurznamen abfragen, verwendet der SQL-Compiler die Metadaten des Index, wie z. B. den Namen jeder Spalte im Indexschlüssel.

### Vorgehensweise:

Sie können einen Kurznamen von der DB2-Steuerzentrale aus oder über die DB2-Befehlszeile registrieren:

- Um diese Task von der DB2-Steuerzentrale aus auszuführen, verwenden Sie den Assistenten für zusammengeschlossene Objekte oder klicken Sie mit der rechten Maustaste den Ordner **Kurznamen** an, und klicken Sie anschließend **Erstellen** an. Verwenden Sie das Discover-Tool zum Identifizieren der Objekte, für die Sie Kurznamen erstellen möchten.
- Um diese Task von der DB2-Befehlszeile aus auszuführen, verwenden Sie die Anweisung CREATE NICKNAME STATEMENT.

Sie können mehrere Kurznamen für dasselbe Datenquellenobjekt angeben.

### Zugehörige Tasks:

- „Registrieren der Kurznamen für Dateien mit Tabellenstruktur“ auf Seite 408
- „Registrieren von Kurznamen für Documentum-Datenquellen“ auf Seite 199
- „Registrieren von Kurznamen für Excel-Datenquellen“ auf Seite 253
- „Registrieren von Kurznamen für BLAST-Datenquellen“ auf Seite 123
- „Registrieren von Kurznamen für XML-Datenquellen“ auf Seite 482
- „Registrieren von Kurznamen für Entrez-Datenquellen“ auf Seite 228
- „Registrieren von Kurznamen für Teradata-Tabellen und -Sichten“ auf Seite 428
- „Registrieren von Kurznamen für BioRS-Datenquellen“ auf Seite 84
- „Registrieren von Kurznamen für DB2-Tabellen und -Sichten“ auf Seite 189
- „Registrieren von Kurznamen für Extended Search-Datenquellen“ auf Seite 274
- „Registrieren von Kurznamen für HMMER-Datenquellen“ auf Seite 302
- „Registrieren von Kurznamen für Informix-Tabellen, -Sichten und -Synonyme“ auf Seite 326
- „Registrieren von Kurznamen für Microsoft SQL Server-Tabellen und Sichten“ auf Seite 344
- „Registrieren von Kurznamen für ODBC-Datenquellentabellen und -sichten“ auf Seite 357
- „Registrieren von Kurznamen für Oracle-Tabellen und -Sichten“ auf Seite 384
- „Registrieren von Kurznamen für Sybase-Tabellen und -Sichten“ auf Seite 400

- „Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen“ auf Seite 148
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442

**Zugehörige Referenzen:**

- „Namenskonventionen für Objekte in zusammengeschlossenen Datenbanken“ auf Seite 25

## Erstellen eines Kurznamens für einen Kurznamen

Es kann vorkommen, dass Sie einen Kurznamen für einen Kurznamen erstellen müssen.

**Vorgehensweise:**

Angenommen, Sie haben einen Server mit zusammengeschlossenen Datenbanken, der AIX® verwendet, und einen Server mit zusammengeschlossenen Datenbanken, der Windows verwendet. Sie möchten von beiden Servern mit zusammengeschlossenen Datenbanken auf ein Excel-Arbeitsblatt zugreifen. Der Wrapper für Excel wird jedoch nur von Servern mit zusammengeschlossenen Datenbanken unterstützt, die Windows verwenden. Führen Sie die folgenden Schritte aus, um von dem Server mit zusammengeschlossenen Datenbanken unter AIX auf das Excel-Arbeitsblatt zuzugreifen:

1. Installieren Sie DB2 Information Integrator auf dem Server mit zusammengeschlossenen Datenbanken, der Windows verwendet.
2. Konfigurieren Sie den Windows-Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Excel-Datenquellen.
3. Erstellen Sie auf dem Windows-Server mit zusammengeschlossenen Datenbanken einen Kurznamen für das Excel-Arbeitsblatt.
4. Installieren Sie DB2 Information Integrator auf dem Server mit zusammengeschlossenen Datenbanken, der AIX verwendet.
5. Konfigurieren Sie den AIX-Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen der DB2-Produktfamilie.
6. Erstellen Sie auf dem AIX-Server mit zusammengeschlossenen Datenbanken einen Kurznamen für den auf dem Windows-Server mit zusammengeschlossenen Datenbanken verwendeten Excel-Kurznamen.

## Angeben von Kurznamenspalten für eine nicht relationale Datenquelle

Für manche nicht relationale Datenquellen muss beim Registrieren eines Kurznamens eine Liste der Spalten angegeben werden. Jede angegebene Spalte wird einem bestimmten Feld, einer bestimmten Spalte oder einem bestimmten Element im Datenquellenobjekt zugeordnet.

Die Wrapper mancher nicht relationaler Datenquellen erfordern eine Gruppe fixierter Eingabe- und Ausgabespalten. Die fixierten Spalten werden automatisch definiert, wenn Sie den Kurznamen registrieren, und werden zum Katalog des Systems zusammengeschlossener Datenbanken hinzugefügt.



### Vorgehensweise:

Um beim Registrieren eines Kurznamens eine Liste der Spalten zu definieren, geben Sie den Spaltennamen und Datentyp an. Sie können auch eine Option für die Kurznamenspalte angeben.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für Datenquellen“ auf Seite 73

---

## Optionale Konfigurationsschritte

Unter Umständen sind einige optionale Schritte auszuführen, wenn Sie den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen konfigurieren.

### Indexspezifikationen:

Definieren Sie eine Indexspezifikation für Objekte, die nicht über einen Index verfügen. Beispielsweise erstellen Sie eine Indexspezifikation, wenn eine Tabelle einen neuen Index generiert oder wenn das Datenquellenobjekt (z. B. eine Sicht) typischerweise keinen Index hat.

### Datentypzuordnungen:

Sie können alternative Datentypzuordnungen für ausschließlich relationale Datenquellen angeben.

Geben Sie alternative Datentypzuordnungen an, anstatt die standardmäßigen Datentypzuordnungen zu verwenden. Sie können eine Zuordnung angeben, die nur für ein bestimmtes Datenquellenobjekt verwendet wird, beispielsweise eine bestimmte Tabelle innerhalb einer Datenbank.

### Funktionszuordnungen:

Sie können Funktionszuordnungen für ausschließlich relationale Datenquellen angeben.

Geben Sie alternative Funktionszuordnungen an, anstatt die Standardfunktionszuordnungen zu verwenden. Diese Vorgehensweise ist besonders dann von Nutzen, DB2<sup>®</sup> eine benutzerdefinierte Funktion in der Datenquelle verwenden soll.

### Zugehörige Konzepte:

- „Datentypzuordnungen in einem System zusammengeschlossener Datenbanken“ in *Systeme zusammengeschlossener Datenbanken*
- „Funktionszuordnungen in einem System zusammengeschlossener Datenbanken“ in *Systeme zusammengeschlossener Datenbanken*
- „Indexspezifikationen in einem System zusammengeschlossener Datenbanken“ in *Systeme zusammengeschlossener Datenbanken*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

---

## Kapitel 6. Konfigurieren des Zugriffs auf BioRS-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in BioRS-Datenquellen gespeichert sind. Der Zugriff auf BioRS-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält folgende Informationen:

- Erläuterungen zu BioRS.
- Liste der auszuführenden Tasks.
- Beispiele zu den benötigten SQL-Anweisungen.
- Liste der dem BioRS-Wrapper zugeordneten Fehlermeldungen.

---

### Was ist BioRS?

BioRS ist ein von Biomax Informatics entwickeltes System zum Abfragen und Abrufen von Daten. Mit Hilfe von BioRS können Sie Informationen aus mehreren Datenquellen (einschließlich Flachdateien und relationalen Datenbanken) abrufen. Allgemein zugängliche Daten wie beispielsweise aus SwissProt und GenBank werden normalerweise als Flachdateien in das BioRS-System heruntergeladen. BioRS kann öffentlich zugängliche Datenquellen und proprietäre Datenquellen (beispielsweise nicht öffentliche Datenbanken, die von Ihrem Unternehmen verwaltet werden) in eine einheitliche Umgebung integrieren.

Nach der Integration einer Datenquelle in das BioRS-System wird die betreffende Datenquelle als *Datenbank* bezeichnet. Die unter einem Datenbankeintrag enthaltenen Elemente werden zusammengefasst als *Schema* bezeichnet. Indexierte Elemente einer Datenbank können in den Funktionen `BIORS.CONTAINS`, `BIORS.CONTAINS_GE` und `BIORS.CONTAINS_LE` verwendet werden. Die BioRS-Funktionen werden in der Klausel `WHERE` der Anweisung `SELECT` angegeben. Auf nicht indexierte Elemente kann in der `SELECT`-Liste sowie in anderen Vergleichselementen in der Klausel `WHERE` verwiesen werden. Nicht indexierte Elemente werden vom Server mit zusammengeschlossenen Datenbanken verarbeitet.

Zwischen Einträgen in Datenbanken können Abhängigkeiten erstellt werden, sodass sich Datenbanken im BioRS-System miteinander verbinden lassen.

BioRS-Datenbanken können über Eltern-Kind-Abhängigkeiten verfügen, das heißt, sie können verschachtelt sein. In einer solchen Abhängigkeit enthält die untergeordnete Datenbank (Kind) ein Element namens `PARENT` vom Datentyp 'Reference' (Verweis). Das Element `PARENT` verweist auf das Element `_ID_` der übergeordneten Datenbank (Elter). Abgesehen davon, dass dieses vordefinierte Element `PARENT` vorhanden ist, enthalten verschachtelte Datenbanken die gleichen Daten wie nicht verschachtelte Datenbanken.

BioRS stellt eine webbasierte Schnittstelle zur Verfügung, mit deren Hilfe Benutzer Abfragen nach Daten in BioRS-Datenbanken ausführen können. Der BioRS-Wrapper verwendet zum Ausführen von Abfragen die gleichen Anwendungsschnittstellen (Application Programming Interfaces, APIs) wie die webbasierte BioRS-Schnittstelle.

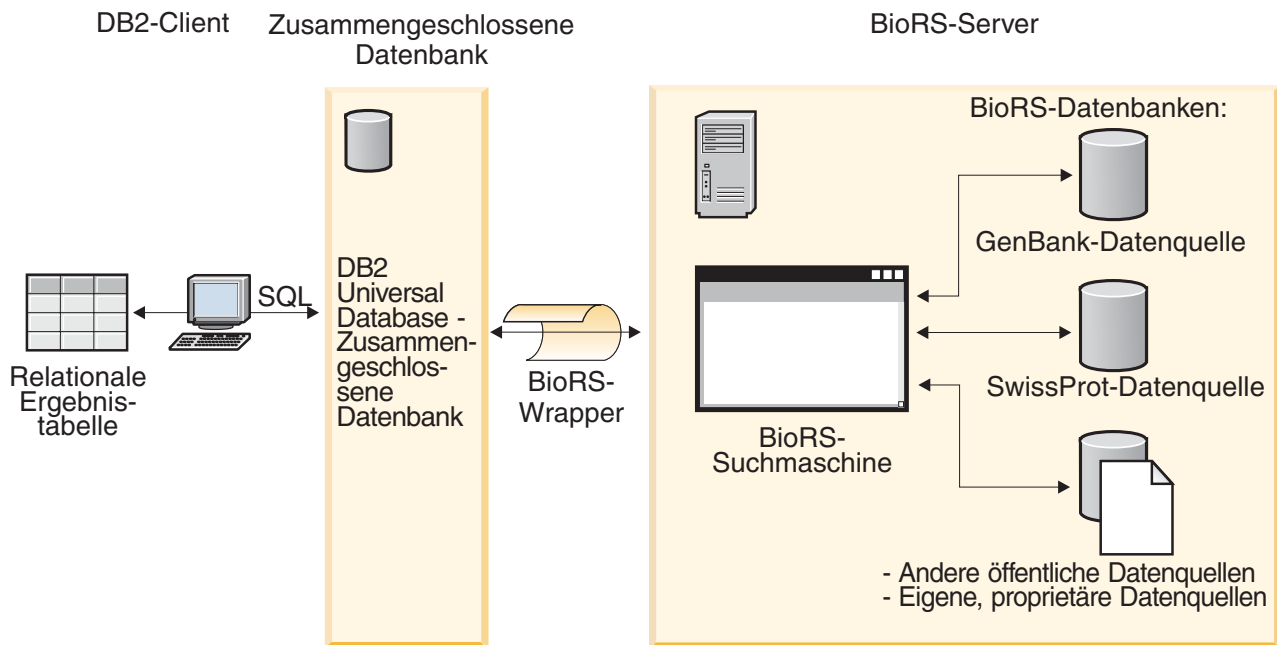


Abbildung 6. Funktionsweise des BioRS-Wrappers

Vom Client aus können Benutzer oder Anwendungen Abfragen mit Hilfe von SQL-Anweisungen übergeben. Anschließend werden die Abfragen an das System zusammengeschlossener Datenbanken gesendet, in dem der BioRS-Wrapper installiert ist. Abhängig vom Aufbau der Abfrage kann diese sowohl von Servern mit zusammengeschlossenen Datenbanken als auch von Ihrem BioRS-Server verarbeitet werden. Der BioRS-Server muss sich nicht auf demselben Computer befinden wie das System zusammengeschlossener Datenbanken. Für jede Abfrage muss das System zusammengeschlossener Datenbanken dem BioRS-Server entsprechende Authentifizierungsinformationen zur Verfügung stellen. Bei diesen Informationen kann es sich entweder um eine Kombination aus Benutzer-ID und Kennwort handeln oder um eine nicht authentifizierte Angabe (normalerweise ein Gastaccount).

Der BioRS-Wrapper funktioniert mit BioRS Version 5.0.14.

Ausführliche Informationen zu BioRS finden Sie auf der Website von Biomax unter: <http://www.biomax.com>

#### Zugehörige Tasks:

- „Hinzufügen von BioRS-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 79

#### Zugehörige Referenzen:

- „BioRS-Wrapper - Beispielabfragen“ auf Seite 93

---

## Hinzufügen von BioRS zu einem Server mit zusammengeschlossenen Datenbanken

### Hinzufügen von BioRS-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf BioRS-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken auf BioRS-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

#### Voraussetzungen:

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.

#### Vorgehensweise:

Führen Sie folgende Schritte aus, um BioRS-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Registrieren Sie angepasste Funktionen für den BioRS-Wrapper.
2. Registrieren Sie den BioRS-Wrapper.
3. Registrieren Sie die BioRS-Serverdefinition.
4. Optional: Erstellen Sie Benutzerzuordnungen.
5. Registrieren Sie Kurznamen für BioRS-Datenbanken.

#### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

#### Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Registrieren von angepassten Funktionen für den BioRS-Wrapper“ auf Seite 80
- „Registrieren des BioRS-Wrappers“ auf Seite 81
- „Registrieren der Serverdefinition für eine BioRS-Datenquelle“ auf Seite 83
- „Registrieren von Benutzerzuordnungen für BioRS-Datenquellen“ auf Seite 83
- „Registrieren von Kurznamen für BioRS-Datenquellen“ auf Seite 84
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43

#### Zugehörige Referenzen:

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## Registrieren von angepassten Funktionen für den BioRS-Wrapper

Die Registrierung von angepassten Funktionen für den BioRS-Wrapper ist Teil der umfangreicheren Task des Hinzufügens von BioRS-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Angepasste Funktionen für den BioRS-Wrapper sind:

- `biors.contains`
- `biors.contains_le`
- `biors.contains_ge`
- `biors.search_term`

### Voraussetzungen:

- Alle angepassten Funktionen für den BioRS-Wrapper müssen mit dem Schemanamen `biors` registriert werden.
- Jede angepasste Funktion muss einmal für jede zusammengeschlossene Datenbank registriert werden, in der der BioRS-Wrapper installiert ist.

### Vorgehensweise:

Um angepasste Funktionen zu registrieren, geben Sie die Anweisung `CREATE FUNCTION` mit den Schlüsselwörtern `AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION` aus.

Der vollständig qualifizierte Name jeder Funktion ist `biors.function_name`.

Im folgenden Beispiel wird eine Version der Funktion `CONTAINS` registriert:

```
CREATE FUNCTION biors.contains (varchar(), varchar())  
  RETURNS INTEGER AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION;
```

Um angepasste Funktionen zu registrieren, verwenden Sie die Beispieldatei `create_function_mappings.ddl`. Diese Beispieldatei befindet sich im Verzeichnis `sqllib/samples/lifesci/biors`. Sie enthält Definitionen für jede angepasste Funktion. Sie können diese DDL-Datei ausführen, um die angepassten Funktionen für jede zusammengeschlossene Datenbank zu registrieren, in der der BioRS-Wrapper installiert ist.

Die nächste Task in dieser Tasksequenz ist das Registrieren des BioRS-Wrappers.

### Zugehörige Tasks:

- „Registrieren des BioRS-Wrappers“ auf Seite 81

### Zugehörige Referenzen:

- „`CREATE FUNCTION (Sourced or Template) statement`“ in *SQL Reference, Volume 2*
- „Angepasste Funktionen und BioRS-Abfragen“ auf Seite 88
- „BioRS-Wrapper - Beispielabfragen“ auf Seite 93
- „Tabelle der angepassten Funktionen - BioRS-Wrapper“ auf Seite 81

## Tabelle der angepassten Funktionen - BioRS-Wrapper

Verwenden Sie die Anweisung CREATE FUNCTION, um die angepassten BioRS-Funktionen zu registrieren.

Die folgende Tabelle listet die vier angepassten BioRS-Funktionen mit Beispielen der Datentypen auf, die Sie beim Registrieren der Funktionen angeben können.

Der erste in der Funktion angegebene Datentyp ist die indexierte Spalte. Der zweite in der Funktion angegebene Datentyp ist der Suchbegriff.

*Tabelle 12. Angepasste Funktionen für den BioRS-Wrapper*

<b>Funktion</b>	<b>Beschreibung</b>
biors.contains (varchar(), varchar()) biors.contains (varchar(), char()) biors.contains (varchar(), date) biors.contains (varchar(), timestamp)	Durchsucht eine indexierte Spalte nach Werten, die dem von Ihnen angegebenen Wert entsprechen (ausgehend von der BioRS-Abfragesemantik). Das erste Argument muss ein Verweis auf die indexierte Spalte sein, und das zweite Argument ist der von Ihnen angegebene Wert.
biors.contains_LE (varchar(), varchar()) biors.contains_LE (varchar(), smallint) biors.contains_LE (varchar(), bigint) biors.contains_LE (varchar(), decimal) biors.contains_LE (varchar(), double) biors.contains_LE (varchar(), real)	Durchsucht eine indexierte Spalte nach Werten, die kleiner als oder genauso groß wie der von Ihnen angegebene Wert sind (ausgehend von der BioRS-Abfragesemantik). Das erste Argument muss ein Verweis auf die indexierte Spalte sein, und das zweite Argument ist der von Ihnen angegebene Wert.
biors.contains_GE (char(), char()) biors.contains_GE (char(), date) biors.contains_GE (char(), timestamp) biors.contains_GE (char(), integer) biors.contains_GE (char(), smallint) biors.contains_GE (clob(), date)	Durchsucht eine indexierte Spalte nach Werten, die größer als oder genauso groß wie der von Ihnen angegebene Wert sind (ausgehend von der BioRS-Abfragesemantik). Das erste Argument muss ein Verweis auf die indexierte Spalte sein, und das zweite Argument ist der von Ihnen angegebene Wert.
biors.search_term (varchar(), varchar()) biors.search_term (varchar(), char()) biors.search_term (char(), varchar()) biors.search_term (char(), char())	Übergibt einen BioRS-Suchbegriff an die BioRS-Suchmaschine.

### Zugehörige Tasks:

- „Registrieren von angepassten Funktionen für den BioRS-Wrapper“ auf Seite 80

## Registrieren des BioRS-Wrappers

Die Registrierung des BioRS-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von BioRS-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf BioRS-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Bibliotheksdatei des Wrappers aus.

Um beispielsweise einen Wrapper mit dem Namen `biors_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER biors_wrapper LIBRARY 'libdb2lsbiors.a';
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung CREATE WRAPPER finden Sie in der Liste der Bibliotheksdateien von BioRS-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinition für den BioRS-Wrapper.

### Zugehörige Referenzen:

- „Bibliotheksdateien für den BioRS-Wrapper“ auf Seite 82
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien für den BioRS-Wrapper

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den BioRS-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, werden 3 Bibliotheksdateien zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden folgende Wrapperbibliotheksdateien zum Verzeichnispfad hinzugefügt: `libdb2lsbiors.a`, `libdb2lsbiorsF.a` und `libdb2lsbiorsU.a`.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 13. Positionen und Dateinamen von BioRS-Wrapperbibliotheken*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2lsbiors.a
Linux	/opt/IBM/db2/V8.1/lib	libdb2lsbiors.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lsbiors.so
Windows	%DB2PATH%\bin	db2lsbiors.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist `C:\Program Files\IBM\SQLLIB`.

### Zugehörige Referenzen:

- „Wrapperbibliotheksdateien“ auf Seite 45



## Registrieren der Serverdefinition für eine BioRS-Datenquelle

Die Registrierung der Serverdefinition für eine BioRS-Datenquelle ist Teil der umfangreicheren Task des Hinzufügens von BioRS zu einem System zusammengesetzter Datenbanken. Nach der Registrierung des Wrappers muss eine entsprechende Serverdefinition registriert werden.

### Vorgehensweise:

Um die BioRS-Serverdefinition in der zusammengesetzten Datenbank zu registrieren, geben Sie die Anweisung CREATE SERVER aus.

Beispiel:

```
CREATE SERVER brs_server WRAPPER wrap_biors OPTIONS(NODE 'biors_server2.com');
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der Benutzerzuordnungen für BioRS-Datenquellen.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für BioRS-Datenquellen“ auf Seite 84

### Zugehörige Referenzen:

- „Anweisungsoptionen für CREATE SERVER - BioRS-Wrapper“ auf Seite 599

## Registrieren von Benutzerzuordnungen für BioRS-Datenquellen

Die Registrierung von Benutzerzuordnungen ist Teil der umfangreicheren Task des Hinzufügens von BioRS zu einem System zusammengesetzter Datenbanken.

Je nachdem, welche Zugriffsmethode für Accounts im BioRS-System verwendet wird, ist es möglicherweise nicht erforderlich, Benutzerzuordnungen zu erstellen.

- Ist der BioRS-Server für Gastzugriff für alle Benutzeraccounts konfiguriert, müssen in DB2 Information Integrator keine Benutzerzuordnungen erstellt werden.
- Ist der BioRS-Server für die Authentifizierung von Benutzeraccounts durch IDs und Kennwörter konfiguriert, müssen Sie in Ihrer zusammengesetzten Datenbank Benutzerzuordnungen für die Accounts erstellen, die mit dem BioRS-Wrapper arbeiten müssen.
- Ist der BioRS-Server für die Verwendung sowohl von Gast- als auch von authentifizierten Benutzeraccounts konfiguriert, müssen Sie in Ihrer zusammengesetzten Datenbank Benutzerzuordnungen für diejenigen authentifizierten Benutzeraccounts erstellen, die mit dem BioRS-Wrapper arbeiten müssen.

Benutzerzuordnungen bieten eine Möglichkeit für die Authentifizierung des Zugriffs von Benutzern oder Anwendungen, die eine BioRS-Datenquelle mit dem BioRS-Wrapper abfragen. Übergibt ein Benutzer oder eine Anwendung eine SQL-Abfrage an einen registrierten BioRS-Kurznamen, ohne dass für den betreffenden Benutzer oder die betreffende Anwendung Benutzerzuordnungen definiert sind, so verwendet der BioRS-Wrapper eine standardmäßige Benutzer-ID mit Kennwort und versucht, Daten vom fernen BioRS-Server abzurufen. Ist für eine abgefragte Datenbank eine Authentifizierung erforderlich, wird unter Umständen eine Fehlermeldung zurückgegeben.



Um sicherzustellen, dass die korrekte Benutzer-ID mit Kennwort an den BioRS-Server übergeben wird, erstellen Sie in Ihrem System zusammenschlossener Datenbanken Benutzerzuordnungen für Benutzer, die für die Suche in BioRS-Datenquellen berechtigt sind. Wenn Sie eine Benutzerzuordnung erstellen, wird das Kennwort in verschlüsseltem Format in einer Katalogtabelle des Systems zusammenschlossener Datenbanken gespeichert.

#### **Vorgehensweise:**

Um BioRS-Benutzerzuordnungen zu registrieren, verwenden Sie die Anweisung CREATE USER MAPPING.

Mit der folgenden Anweisung CREATE USER MAPPING beispielsweise wird der Benutzer Charlie dem Benutzer Charlene auf dem Server Biors\_Server1 zugeordnet.

```
CREATE USER MAPPING FOR Charlie SERVER Biors_Server1
OPTIONS(REMOTE_AUTHID 'Charlene', REMOTE_PASSWORD 'Charlene_pw');
```

Sie können auch eine eigene Benutzerzuordnung definieren. In folgendem Beispiel ist USER ein Schlüsselwort, das den aktuellen Benutzer identifiziert, und kein Benutzername von USER.

```
CREATE USER MAPPING FOR USER SERVER Biors_Server1
OPTIONS(REMOTE_AUTHID 'Yudong', REMOTE_PASSWORD 'Yudong_pw')
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für BioRS-Datenquellen.

#### **Zugehörige Tasks:**

- „Registrieren von Kurznamen für BioRS-Datenquellen“ auf Seite 84

#### **Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „Anweisungsoptionen für CREATE USER MAPPING - BioRS-Wrapper“ auf Seite 600

## **Registrieren von Kurznamen für BioRS-Datenquellen**

Das Registrieren von Kurznamen für BioRS-Datenquellen ist Teil der umfangreichen Task des Hinzufügens von BioRS zu einem Server mit zusammenschlossenen Datenbanken.

Nach der Registrierung einer Serverdefinition muss ein Kurzname für jede BioRS-Datenquelle registriert werden, auf die zugegriffen werden soll. Wenn Sie in einer Abfrage auf eine BioRS-Datenquelle verweisen, verwenden Sie den Kurznamen.

Nach der Integration einer Datenquelle in das BioRS-System wird diese in BioRS als *Datenbank* bzw. als *Databank* bezeichnet. Datenbanken in BioRS sind mit Kurznamen in einem System zusammenschlossener Datenbanken gleichzusetzen.

#### **Voraussetzungen:**

- Falls ein BioRS-Datenbankname nicht mit der für die Anweisung CREATE NICKNAME erforderlichen Syntax übereinstimmt, müssen Sie beim Registrieren des Kurznamens die Kurznamenoption REMOTE\_OBJECT verwenden.

- Falls ein BioRS-Elementname nicht mit der für die Anweisung CREATE NICKNAME erforderlichen Syntax übereinstimmt, müssen Sie beim Registrieren des Kurznamens die Spaltenoption ELEMENT\_NAME verwenden.

### **Einschränkungen:**

Verwenden Sie das BioRS-Element 'AllText' nicht als erste Spalte für einen Kurznamen. In allen anderen Spaltenpositionen (beispielsweise als zweite oder dritte Spalte) kann das BioRS-Element 'AllText' verwendet werden.

### **Vorgehensweise:**

Um einen BioRS-Kurznamen zu registrieren, verwenden Sie die Anweisung CREATE NICKNAME.

Wenn Sie einen BioRS-Kurznamen erstellen, definieren Sie eine Liste von Kurznamenspalten. Die angegebenen Kurznamenspalten müssen den Elementen eines bestimmten BioRS-Datenbankformats entsprechen. BioRS definiert fünf mögliche Datentypen für Elemente: Text, Number (Nummer), Date (Datum), Author (Autor) und Reference (Verweis). BioRS-Datentypen können nur den DB2-Datentypen CHAR, CLOB oder VARCHAR zugeordnet werden.

Am einfachsten lässt sich ein Kurzname für eine BioRS-Datenbank registrieren, wenn der Kurzname denselben Namen erhält wie die BioRS-Datenbank. Beispiel:

```
CREATE NICKNAME SwissProt
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR(128),
  ENTRYDATE VARCHAR (64))
FOR SERVER brs_server;
```

Die BioRS-Datenbank SwissProt ist der Name des Kurznamens.

Durch die Verwendung dieser einfachen Syntax für CREATE NICKNAME werden die Kurznamen auf eine einzige Kurznamenfamilie pro DB2-Schema begrenzt. Sie haben beispielsweise zwei Datenbanken mit einer Elter-Kind-Beziehung. Diese Datenbanken sind SWISSPROT und SPFEAT. Sie bilden eine Familie. Wenn Sie die Standardsyntax für die Anweisung CREATE NICKNAME verwenden, haben Sie einen Kurznamen (SWISSPROT) für die Datenbank SWISSPROT, und einen Kurznamen (SPFEAT) für die Datenbank SPFEAT. Um mehr als einen Kurznamen für SWISSPROT in das Schema einzuschließen, müssen Sie die Option REMOTE\_OBJECT verwenden.

Die Kurznamenoption REMOTE\_OBJECT gibt den Namen der BioRS-Datenbank an, die dem Kurznamen zugeordnet ist. Der in der Option REMOTE\_OBJECT angegebene Name legt das Schema und die BioRS-Datenbank für den Kurznamen fest. Außerdem gibt die Option REMOTE\_OBJECT die Abhängigkeit zwischen dem Kurznamen und anderen Kurznamen an.

Das folgende Beispiel enthält die gleiche Gruppe von Kurznamenmerkmalen wie das vorherige Beispiel. In diesem Beispiel wird der Name des Kurznamens jedoch geändert, und es wird die Option REMOTE\_OBJECT verwendet, um die BioRS-Datenbank anzugeben, für die der Kurzname definiert wird:

```

CREATE NICKNAME NewSP
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR(128),
  ENTRYDATE VARCHAR (64))
FOR SERVER brs_server
  OPTIONS (REMOTE_OBJECT 'SwissProt');

```

Wiederholen Sie diesen Schritt für jede BioRS-Datenbank, für die Sie einen Kurznamen erstellen möchten.

Dies war der letzte Schritt in dieser Tasksequenz.

#### Zugehörige Konzepte:

- „Statistische BioRS-Daten“ auf Seite 99

#### Zugehörige Tasks:

- „Aktualisieren der Kardinalitätsstatistiken von BioRS-Kurznamen“ auf Seite 100
- „Angaben von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75

#### Zugehörige Referenzen:

- „Das BioRS-Element 'AllText'“ auf Seite 92
- „Anweisung CREATE NICKNAME - Beispiele für den BioRS-Wrapper“ auf Seite 86
- „Anweisungssyntax für CREATE NICKNAME - BioRS-Wrapper“ auf Seite 601

## Anweisung CREATE NICKNAME - Beispiele für den BioRS-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE NICKNAME verwendet wird, um Kurznamen für den BioRS-Wrapper zu registrieren.

#### Beispiel 1:

Das folgende Beispiel zeigt, wie ein Kurzname für eine ferne BioRS-Datenbank erstellt wird, die nicht der Syntax von DB2 Information Integrator entspricht:

```

CREATE NICKNAME SwissFT
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR (128),
  ENTRYDATE VARCHAR (64),
  FtLength VARCHAR (16),
  FOR SERVER biors1
  OPTIONS (REMOTE_OBJECT 'SwissProt.Features');

```

Der Name dieses Kurznamens lautet 'SwissFT'. Die Tabellenspalten sind ID, ALLTEXT, ENTRYDATE und FtLength. Die Spaltenoption ELEMENT\_NAME ist für die ID-Spalte angegeben. Die Option ELEMENT\_NAME muss angegeben werden, wenn der Name eines BioRS-Elements nicht der gültigen DB2-Syntax zusammengesetzter Systeme für Spaltennamen entspricht. In diesem Beispiel entspricht das BioRS-Element \_ID\_ zwar der DB2-Syntax zusammengesetzter Systeme, doch ist \_ID\_ ein möglicherweise verwirrender Name für Benutzer von DB2 Information Integrator. Der Name ID ist einfach und leicht verständlich. Im Allgemeinen wird die Option ELEMENT\_NAME unter folgenden Umständen verwendet:

- Wenn der Name eines BioRS-Elements nicht der gültigen DB2-Syntax zusammengesetzter Systeme entspricht;

- Wenn die Groß-/Kleinschreibung des Namens eines BioRS-Elements nicht den erstellten DB2-Standards des Systems zusammengeschlossener Datenbanken entspricht;
- Wenn der Name eines BioRS-Elements für Benutzer von DB2 Information Integrator möglicherweise nicht eindeutig ist.

Außerdem wird mit der Option REMOTE\_OBJECT der Name der BioRS-Datenbank angegeben, dem der Kurzname entspricht. Die Option REMOTE\_OBJECT muss angegeben werden, wenn der Name einer BioRS-Datenbank nicht der gültigen DB2-Syntax zusammengeschlossener Systeme entspricht. In diesem Beispiel entspricht der Datenbankname "SwissProt.Features" nicht der gültigen DB2-Syntax zusammengeschlossener Systeme. Im Allgemeinen wird die Option REMOTE\_OBJECT unter folgenden Umständen verwendet:

- Wenn die Groß-/Kleinschreibung des Namens einer BioRS-Datenbank nicht den erstellten DB2-Standards des Systems zusammengeschlossener Datenbanken entspricht;
- Wenn der Name einer BioRS-Datenbank nicht der gültigen DB2-Syntax zusammengeschlossener Systeme entspricht;
- Wenn der Name einer BioRS-Datenbank für Benutzer von DB2 Information Integrator möglicherweise nicht eindeutig ist.

### Beispiel 2:

Das folgende Beispiel zeigt, wie ein Kurzname für eine Tabelle erstellt wird, die eine BioRS-Datenbank verwendet, die mit einer anderen BioRS-Datenbank verbunden ist:

```
CREATE NICKNAME SwissFT2
  (ID VARCHAR(32) OPTIONS (ELEMENT_NAME '_ID_'),
  ALLTEXT VARCHAR (1200),
  FtKey VARCHAR (32),
  FtLength VARCHAR (64),
  FtDescription VARCHAR (128),
  Parent VARCHAR (32) OPTIONS (REFERENCED_OBJECT 'SwissProt'))
FOR SERVER biors1
OPTIONS (REMOTE_OBJECT 'SwissProt.Features');
```

Der Name dieses Kurznamens lautet 'SwissFT2'. Die Tabellenspalten sind ID, ALLTEXT, FtKey, FtLength, FtDescription und Parent. Die Spaltenoption ELEMENT\_NAME ist für die ID-Spalte angegeben. Mit der Option REMOTE\_OBJECT wird der Name der BioRS-Datenbank angegeben, dem der Kurzname entspricht.

Die Spalte 'Parent' verwendet außerdem die Option REFERENCED\_OBJECT. Sie müssen diese Option für Spalten angeben, die BioRS-Elementen vom Datentyp 'Reference' (Verweis) entsprechen. Die Option REFERENCED\_OBJECT gibt den Namen der BioRS-Datenbank an, auf die die Spalte verweist. In diesem Fall verweist das Element 'Parent' auf die BioRS-Datenbank 'SwissProt'.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für BioRS-Datenquellen“ auf Seite 84

### Zugehörige Referenzen:

- „Anweisungssyntax für CREATE NICKNAME - BioRS-Wrapper“ auf Seite 601

## Abfragen und angepasste Funktionen für BioRS-Datenquellen

### Angepasste Funktionen und BioRS-Abfragen

Die Umgebung zusammenschlossener Datenbanken verwendet zwei Steuerkomponenten für die Abfrage. Für den BioRS-Wrapper sind diese Steuerkomponenten 'DB2 Universal Database' und 'BioRS'. Mit Hilfe der folgenden vier angepassten BioRS-Funktionen können Sie angeben, dass Vergleichselemente im Pushdown-Modus an die BioRS-Steuerkomponente gesendet werden:

- BIOR.S.CONTAINS
- BIOR.S.CONTAINS\_LE
- BIOR.S.CONTAINS\_GE
- BIOR.S.SEARCH\_TERM

Diese vier angepassten Funktionen werden im BioRS-Schema registriert. Sie müssen das BioRS-Schema verwenden, um auf die Funktionen zu verweisen.

Die angepassten Funktionen BIOR.S.CONTAINS, BIOR.S.CONTAINS\_LE und BIOR.S.CONTAINS\_GE erfordern ein Argument für die Suchbegriffsspalte und ein Argument für den Abfragetext. Das folgende Beispiel zeigt eine Anweisung BIOR.S.CONTAINS:

```
BIOR.S.CONTAINS (<suchbegriffsspalte>,<abfragebegriff>)
```

Der Wert des Arguments für die Suchbegriffsspalte muss auf eine indizierte BioRS-Spalte verweisen. Bei Verwendung einer nicht indizierten Spalte wird Fehlermeldung SQL30090N zurückgegeben ("Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig").

Der Wert des Arguments für den Abfragebegriff kann nur ein Literal, eine Hostvariable oder ein Spaltenbezug sein. Arithmetische Verkettungen und Verkettungen von Zeichenfolgen können nicht verwendet werden. Außerdem darf der Wert des Arguments für den Abfragebegriff nicht NULL sein, und zwar auch dann nicht, wenn die verwendete Suchbegriffsspalte so definiert ist, dass Nullwerte zulässig sind.

Die Groß-/Kleinschreibung ist beim Argument für den Abfragebegriff unerheblich.

Welche Datentypen und Formate des Arguments für den Abfragebegriff gültig sind, hängt vom BioRS-Datentyp der verwendeten Suchbegriffsspalte ab. BioRS definiert fünf mögliche Datentypen: Text, Author (Autor), Date (Datum), Number (Zahl) und Reference (Verweis). Tabelle 14 enthält eine Liste der BioRS-Datentypen und der gültigen Abfragebegriffe für die einzelnen Datentypen.

*Tabelle 14. BioRS-Datentypen und gültige Abfragebegriffe für angepasste Funktionen*

Datentyp der Suchbegriffsspalte	Gültiger Abfragebegriff	Format
Text	VARCHAR() oder CHAR()	BioRS-Textbegriff, einschließlich Platzhalterzeichen

Tabelle 14. BioRS-Datentypen und gültige Abfragebegriffe für angepasste Funktionen (Forts.)

Datentyp der Suchbegriffsspalte	Gültiger Abfragebegriff	Format
Author	VARCHAR() oder CHAR()	BioRS-Autorenverweis im Format " <code>&lt;nachname&gt;, &lt;initialen&gt;</code> ". " <code>&lt;nachname&gt;</code> " ist der Nachname des Autors. " <code>&lt;initialen&gt;</code> " sind die Initialen des Autors ohne Punkte. Leerzeichen zwischen dem Komma und den Initialen sind akzeptabel.  Alternativ kann <code>&lt;nachname&gt;</code> auch allein angegeben werden, ohne das Komma oder die Initialen.
Date	VARCHAR(), CHAR(), DATE oder TIMESTAMP	Bei Zeichenfolge: Datum im DB2-Format, <code>jjjj/mm/tt</code> .
Number	VARCHAR() oder CHAR(), INTEGER, SMALLINT, BIGINT REAL, DOUBLE, DECIMAL	Zahlen im DB2-Format.
Reference	VARCHAR() oder CHAR()	BioRS-Textbegriff.

Alle anderen Kombinationen aus Suchbegriffsspalten mit BioRS-Datentypen und Argumenten für Abfragebegriffe resultieren in Fehlernachricht SQL30090N ("Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig"). Sie können lediglich die in Tabelle 14 auf Seite 88 aufgeführten Kombinationen verwenden.

Das Abfragebegriffsargument für Suchbegriffsspalten mit dem Datentyp 'Text', 'Author' und 'Reference' muss dem Muster der BioRS-Abfragesprache entsprechen. In BioRS können Argumente für Abfragebegriffe aus alphanumerischen Zeichenfolgen und Platzhalterzeichen bestehen. Die Funktion BIOR.S.CONTAINS unterstützt zwei Platzhalterzeichen: ? (Fragezeichen) und \* (Stern).

Das Platzhalterzeichen ? entspricht einem einzelnen Zeichen. Beispiel: Das Vergleichselement BIOR.S.CONTAINS (description, 'bacteri?')=1 entspricht den Begriffen 'bacteria' (Bakterien), aber nicht dem Begriff 'bacterial' (bakteriell).

Das Platzhalterzeichen \* entspricht keinem oder mehreren Zeichen. Beispiel: Das Vergleichselement BIOR.S.CONTAINS (description, 'bacteri\*')=1 entspricht den Begriffen 'bacteri', 'bacteria' und 'bacterial'.

Die BioRS-Dokumentation enthält ausführliche Informationen zu den Mustern der BioRS-Abfragesprache.

Die Funktion BIOR.S.CONTAINS kann für alle BioRS-Spalentypen angegeben werden.

Die angepassten Funktionen BIOR.S.CONTAINS\_GE und BIOR.S.CONTAINS\_LE können lediglich für Spalten angegeben werden, deren zugrunde liegender BioRS-Datentyp 'Number' oder 'Date' ist. Die Funktion BIOR.S.CONTAINS\_GE wählt Zeilen aus, in denen die Spalte einen Wert enthält, der größer-gleich dem Wert ist, der vom Argument für den Abfragebegriff dargestellt wird. Die Funktion BIOR.S.CONTAINS\_LE wählt Zeilen aus, in denen die Spalte einen Wert enthält, der kleiner-gleich dem Wert ist, der vom Argument für den Abfragebegriff dargestellt wird.

Die Funktionen `BIORS.CONTAINS`, `BIORS.CONTAINS_GE` und `BIORS.CONTAINS_LE` geben ein Ergebnis aus ganzen Zahlen zurück. Wird eine der drei `CONTAINS`-Funktionen in einem Vergleichselement verwendet, muss der Rückgabewert mit dem Wert 1 unter Verwendung des Operators `=` oder `<>` verglichen werden.

Beispiel:

```
SELECT * FROM s.MySP WHERE BIORS.CONTAINS (s.AllText, 'muscus') = 1;
```

Ein Ausdruck im Format `NOT (BioRS.Contains (col,value) = 1)` entspricht `BioRS.CONTAINS (col,value) <> 1`.

Durch Absetzen der Funktion `BIORS.SEARCH_TERM` können Sie Abfragen ausführen, die ansonsten unter Umständen nicht möglich wären. Mit Hilfe dieser Funktion können Sie einen Suchbegriff im BioRS-Format angeben. Die Funktion `BIORS.SEARCH_TERM` erfordert zwei Argumente. Das erste Argument ist ein Verweis auf die Spalte `_ID_` des Kurznamens, auf den der Begriff angewendet werden soll. Das zweite Argument ist eine Zeichenfolge, die den Begriff ohne einen Datenbanknamen enthält.

Im folgenden Beispiel werden alle Spalten für Einträge in der Datenbank `MyEMBL` ausgewählt, in denen das Element `'SeqLength'` einen Wert größer-gleich 100 enthält:

```
SELECT * FROM MyEMBL s WHERE  
BIORS.SEARCH_TERM (s.ID, '[SeqLength GREATER number:100;]') = 1;
```

In folgendem Beispiel wird die Spalte `'MolWeight'` aus dem Kurznamen `'Swiss'` ausgewählt, in der der Wert des Elements `'MolWeight'` größer-gleich 100368 ist:

```
SELECT s.molweight FROM Swiss s WHERE  
BIORS.SEARCH_TERM (s.ID, '[MolWeight GREATER number:100368;]') = 1;
```

#### **Zugehörige Konzepte:**

- „Pushdown-Analyse“ in *Systeme zusammenschlossener Datenbanken*
- „Richtlinien zur Leistungsoptimierung des BioRS-Wrappers“ auf Seite 98
- „Vergleichselemente für Verknüpfungen mit Gleichheitsattributen für den BioRS-Wrapper“ auf Seite 90

#### **Zugehörige Tasks:**

- „Registrieren von angepassten Funktionen für den BioRS-Wrapper“ auf Seite 80

#### **Zugehörige Referenzen:**

- „BioRS-Wrapper - Beispielabfragen“ auf Seite 93
- „Tabelle der angepassten Funktionen - BioRS-Wrapper“ auf Seite 81

## **Vergleichselemente für Verknüpfungen mit Gleichheitsattributen für den BioRS-Wrapper**

Bei Verwendung der angepassten BioRS-Funktionen müssen Sie Vergleichselemente für die BioRS-Steuerkomponente angeben, mit einer Ausnahme. Diese Ausnahme besteht bei der Ausführung von Verknüpfungsoperationen mit Gleichheitsattributen während einer Abfrage. Eine *Verknüpfungsoperation* (`JOIN`) umfasst das Abrufen von Daten aus zwei oder mehreren Tabellen auf der Grundlage übereinstimmender Spaltenwerte. Eine *Verknüpfung mit Gleichheitsattributen* ist eine Verknüpfungsoperation, bei der die Verknüpfungsbedingung im Format `'ausdruck = ausdruck'` vorliegt. Bei BioRS-Abfragen müssen Bedingungen für Verknüpfungen



mit Gleichheitsattributen das Element `_ID_` einer Datenbank und ein Element vom Typ 'Reference' (Verweis) einer anderen Datenbank enthalten.

### Beispiel:

Dieses Beispiel zeigt Definitionen von Beispielkurznamen sowie eine Abfrage mit einer Verknüpfung mit Gleichheitsattribut, in der die Beispielkurznamen verwendet werden.

Sie möchten zwei BioRS-Datenbanken abfragen: 'SwissProt' und 'SwissProt.features'. Die Datenbank 'SwissProt.features' ist ein untergeordnetes Element (Kind) der Datenbank 'SwissProt' und enthält ein Element namens PARENT. Das Element PARENT enthält Verweise auf Einträge, die durch das SwissProt-Element `_ID_` identifiziert werden. Für die beiden Datenbanken werden zwei Kurznamen-Definitionen registriert.

#### Kurznamen-Definition 1:

```
CREATE NICKNAME tc600sprot (  
  ID          VARCHAR (32) OPTIONS (ELEMENT_NAME '_ID_'),  
  AllText     VARCHAR (128),  
  EntryDate   VARCHAR (128),  
  Update      VARCHAR (128),  
  Description  VARCHAR (1200),  
  Crossreference VARCHAR (32),  
  Authors     VARCHAR (256),  
  Journal     VARCHAR (256),  
  JournalIssue VARCHAR (64) OPTIONS (IS_INDEXED 'N'),  
  PublicationYear VARCHAR (1024),  
  Gene        VARCHAR (20) OPTIONS (IS_INDEXED 'Y'),  
  Remarks     VARCHAR (1200),  
  RemarkType  CHAR (20),  
  CatalyticActivity VARCHAR (20),  
  CoFactor    VARCHAR (64),  
  Disease     VARCHAR (128),  
  Function    VARCHAR (128),  
  Pathway     VARCHAR (128),  
  Similarity  VARCHAR (128),  
  Complex     VARCHAR (64),  
  FtKey       VARCHAR (32),  
  FtDescription VARCHAR (128),  
  FtLength    VARCHAR (256),  
  MolWeight   VARCHAR (64),  
  ProteinLen  VARCHAR (32) OPTIONS (ELEMENT_NAME 'Protein_length'),  
  Sequence    CLOB,  
  AccNumber   VARCHAR (32),  
  Taxonomy    VARCHAR (128),  
  Organelle   VARCHAR (128),  
  Organism    VARCHAR (128),  
  Keywords    VARCHAR (1200),  
  Localization VARCHAR (128),  
  FtKey_count VARCHAR (32)) FOR SERVER biors_server_600  
  OPTIONS (REMOTE_OBJECT 'SwissProt');
```

#### Kurznamen-Definition 2:

```
CREATE NICKNAME tc600feat (  
  ID          VARCHAR (32) OPTIONS (ELEMENT_NAME '_ID_'),  
  AllText     VARCHAR (1200),  
  FtKey       VARCHAR (32),  
  FtLength    VARCHAR (64),  
  FtDescription VARCHAR (128),  
  Parent      VARCHAR (32) OPTIONS (REFERENCED_OBJECT 'SwissProt'))  
  FOR SERVER biors_server_600 OPTIONS (REMOTE_OBJECT 'SwissProt.features');
```



Die folgende Abfrage verweist in einer Verknüpfung mit Gleichheitsattribut auf beide Kurznamen:

```
SELECT s.ID, f.ID, f.FtKey FROM tc600sprot s, tc600feat f
WHERE BioRS.CONTAINS (s.AllText, 'anopheles') = 1
AND BioRS.CONTAINS (s.PublicationYear, 1997) = 1
AND BioRS.CONTAINS (f.FtKey, 'signal') = 1
AND f.Parent = s.ID;
```

In der vorstehenden Abfrage werden zwei Vergleichselemente auf den Kurznamen 'tc600sprot' (Datenbank 'SwissProt') angewendet. Diese beiden Vergleichselemente filtern die Zeilen heraus, die den Begriff 'anopheles' enthalten und deren Publikationsjahr (PublicationYear) 1997 ist. Ein Vergleichselement wird auf den Kurznamen 'tc600feat' (Datenbank 'SwissProt.features') angewendet und filtert die Zeilen heraus, deren Element 'FtKey' den Begriff 'signal' enthält. Beide Kurznamen werden mit Hilfe der Bedingung 'term f.Parent = s.ID' miteinander verknüpft. Die endgültige Ergebnismenge enthält nur diejenigen Zeilen, die diese Kriterien erfüllen und in denen die Merkmalseinträge auf einen übereinstimmenden Eintrag in der Datenbank 'SwissProt' verweisen.

#### **Zugehörige Konzepte:**

- „Richtlinien zur Leistungsoptimierung des BioRS-Wrappers“ auf Seite 98

#### **Zugehörige Referenzen:**

- „Angepasste Funktionen und BioRS-Abfragen“ auf Seite 88
- „BioRS-Wrapper - Beispielabfragen“ auf Seite 93

## **Das BioRS-Element 'AllText'**

Jede Datenbank im BioRS-System enthält ein Element namens 'AllText'. Das Element 'AllText' ist ein indexiertes Element, das von BioRS automatisch für alle Datenbanken erstellt wird. Mit dem Element 'AllText' können Sie den gesamten Text eines Eintrags durchsuchen, nicht nur bestimmte indexierte Elemente. Die Suche nach dem Begriff muscus beispielsweise kann Einträge zurückgeben, in denen das Wort muscus im Titel ('title'), in der Kurzdarstellung ('abstract'), in der Beschreibung ('description') oder im Organismus ('organism') vorkommt.

Um das Element 'AllText' in einer DB2 Information Integrator-Abfrage verwenden zu können, müssen Sie das Element 'AllText' einer Kurznamenspalte zuordnen. Wenn Sie in der Anweisung CREATE NICKNAME Spalten angeben, ordnen Sie das Element 'AllText' einer Kurznamenspalte zu. Eine zu dem Element 'AllText' zugeordnete Kurznamenspalte gibt in SELECT-Anweisungen einen Wert NULL zurück. Wenn Sie eine Spalte als Element 'AllText' angeben, darf dies nicht die in einer Anweisung CREATE NICKNAME erste deklarierte Spalte sein.

Nachdem dieses Element einer Kurznamenspalte ordnungsgemäß zugeordnet wurde, können Sie die betreffende Spalte beim Aufruf einer angepassten CONTAINS-Funktion verwenden.

#### **Zugehörige Tasks:**

- „Registrieren von Kurznamen für BioRS-Datenquellen“ auf Seite 84

#### **Zugehörige Referenzen:**

- „BioRS-Wrapper - Beispielabfragen“ auf Seite 93

## BioRS-Wrapper - Beispielabfragen

Dieser Abschnitt enthält mehrere Beispielabfragen, in denen die Kurznamen 'swiss' und 'swissft' verwendet werden.

Der Kurzname 'swiss' wurde mit Hilfe der folgenden Anweisung CREATE NICKNAME registriert:

```
CREATE NICKNAME swiss
(
  ID          CHAR (30) OPTIONS (ELEMENT_NAME '_ID_'),
  EntryDate   VARCHAR (15),
  Update      CLOB (15),
  Description CLOB (15),
  Crossreference CLOB (15),
  Authors     CLOB (15),
  Journal     VARCHAR (15),
  JournalIssue VARCHAR (15),
  PublicationYear CLOB (15),
  PublicationTitle CLOB (15),
  Gene        CLOB (15),
  Remarks     CLOB (15),
  RemarkType  VARCHAR (15),
  CatalyticActivity VARCHAR (15),
  CoFactor    VARCHAR (15),
  Disease     VARCHAR (15),
  Function    CLOB (15),
  Pathway     VARCHAR (15),
  Similarity  CLOB (15),
  Complex     VARCHAR (15),
  FtKey       VARCHAR (15),
  FtDescription CLOB (15),
  FtLength    VARCHAR (15),
  MolWeight   CHAR (15),
  Protein_Length VARCHAR (15),
  Sequence    CLOB (15),
  AccNumber   VARCHAR (15),
  Taxonomy    CLOB (15),
  Organelle   VARCHAR (15),
  Organism    VARCHAR (15),
  Keywords    VARCHAR (15),
  Localization VARCHAR (15),
  FtKey_count VARCHAR (15),
  AllText     CLOB (15)
)
FOR SERVER biors_server
  OPTIONS (REMOTE_OBJECT 'swissprot');
```

Der Kurzname 'swissft' wurde mit Hilfe der folgenden Anweisung CREATE NICKNAME registriert:

```
CREATE NICKNAME swissft
(
  ID          VARCHAR (30) OPTIONS (ELEMENT_NAME '_ID_'),
  FtKey       VARCHAR (15),
  FtLength    VARCHAR (15),
  FtDescription VARCHAR (15),
  Parent      VARCHAR (30) OPTIONS (REFERENCED_OBJECT 'swissprot'),
  AllText     CLOB (15)
)
FOR SERVER biors_server
  OPTIONS (REMOTE_OBJECT 'swissprot.features');
```

Die Abfragen und Ergebnisse in Tabelle 15 zeigen, wie Sie Ihre Abfragen strukturieren können, um die Verteilung der Verarbeitungslast zwischen dem System zusammengesetzter Datenbanken und dem BioRS-Server zu optimieren.

*Tabelle 15. Beispiele verschiedener Abfragen mit identischen Ergebnissen*

Abfrage	Ergebnis
select s.id from Swiss s where biors.CONTAINS(s.id, '100K_RAT') = 1 fetch first 3 rows only	ID----- 100K_RAT  1 Satz/Sätze ausgewählt.
select s.id from Swiss s where s.id LIKE '%100K_RAT%' fetch first 3 rows only	ID----- 100K_RAT  1 Satz/Sätze ausgewählt.

Beide Abfragen in Tabelle 15 liefern die gleichen Ergebnisse. Die erste Abfrage wird jedoch viel schneller verarbeitet als die zweite. Die erste Abfrage verwendet die Funktion BIOR.SCONTAINS zur Angabe des Eingabevergleichselements. Demzufolge wählt BioRS die Daten in der Datenbank 'swissprot' aus und übergibt die ausgewählten Daten an DB2 Information Integrator. In der zweiten Abfrage wird das Eingabevergleichselement LIKE direkt im Kurznamen 'Swiss' angegeben. Demzufolge leitet BioRS die gesamte Datenbank 'swissprot' an DB2 Information Integrator weiter. Nachdem der Inhalt der Datenbank weitergeleitet wurde, wählt DB2 Information Integrator die Daten aus.

Die Abfragen und Ergebnisse in Tabelle 16 zeigen die Verwendung von Platzhalterzeichen in der Funktion BIOR.SCONTAINS. Alle Abfrageergebnisse in Tabelle 16 sind identisch, obwohl unterschiedliche Platzhalterzeichen verwendet werden.

*Tabelle 16. Beispielabfragen mit Platzhalterzeichen in der Funktion BIOR.SCONTAINS*

Abfrage	Ergebnis
select s.crossreference from Swiss s where biors.CONTAINS(s.crossreference, 'MEDLINE') = 1 fetch first 3 rows only	CROSSREFERENCE ----- NCBI_TaxID=1011 NCBI_TaxID=5875 NCBI_TaxID=4081  3 Satz/Sätze ausgewählt.
select s.crossreference from Swiss s where biors.CONTAINS(s.crossreference, '?ED?IN?') = 1 fetch first 3 rows only	CROSSREFERENCE ----- NCBI_TaxID=1011 NCBI_TaxID=5875 NCBI_TaxID=4081  3 Satz/Sätze ausgewählt.
select s.crossreference from Swiss s where biors.CONTAINS(s.crossreference, '*D*N*') = 1 fetch first 3 rows only	CROSSREFERENCE ----- NCBI_TaxID=1011 NCBI_TaxID=5875 NCBI_TaxID=4081  3 Satz/Sätze ausgewählt.

Die Abfragen und Ergebnisse in Tabelle 17 auf Seite 95 zeigen, wie mit Hilfe der Funktion BIOR.SCONTAINS auf Daten in BioRS-Elementen vom Datentyp 'Author' zugegriffen werden kann.

Die Syntax aller Abfragen in Tabelle 17 ist fast identisch. Der einzige Unterschied besteht darin, dass die Vornamensinitiale im Abfragebegriff entweder vorhanden ist oder nicht, und dass die Anzahl der Leerzeichen zwischen dem Nachnamen und der Vornamensinitiale variiert.

*Tabelle 17. Beispielabfragen zum Zugriff auf BioRS-Spalten vom Datentyp 'Author'*

Abfrage	Ergebnis
select s.authors from Swiss s where biors.CONTAINS(s.authors, 'Mueller') = 1 fetch first 3 rows only	AUTHORS ----- Mueller D. Rehb Mayer K.F.X. Sc Zemmour J. Litt  3 Satz/Sätze ausgewählt.
select s.authors from Swiss s where biors.CONTAINS(s.authors, 'Mueller,D') = 1 fetch first 3 rows only	AUTHORS -----  0 Satz/Sätze ausgewählt.
select s.authors from Swiss s where biors.CONTAINS(s.authors, 'Mueller ,D') = 1 fetch first 3 rows only	AUTHORS -----  0 Satz/Sätze ausgewählt.
select s.authors from Swiss s where biors.CONTAINS(s.authors, 'Mueller, D') = 1 fetch first 3 rows only	AUTHORS ----- Mueller D. Rehb Zou P.J. Borovo Davies J.D. Mue  3 Satz/Sätze ausgewählt.

Die Abfragen und Ergebnisse in Tabelle 18 zeigen, wie mit Hilfe der Funktion BIORS.CONTAINS auf Daten in BioRS-Elementen vom Datentyp 'Date' zugegriffen werden kann.

Wenn ein BioRS-Feld vom Typ 'Date' eine Folge von Datumsangaben enthält, können die Ergebnisse zusätzliche Informationen enthalten, wie aus dem zweiten Beispiel in Tabelle 18 ersichtlich ist. BioRS-Elemente vom Datentyp 'Numeric' ('Date' und 'Number') können mehrere Werte enthalten. Daher können die Ergebnisse von Abfragen, die für BioRS-Elemente vom Typ 'Date' oder 'Number' ausgeführt werden, ebenfalls mehrere Werte enthalten. Mehrere Werte werden stets durch Leerzeichen getrennt.

*Tabelle 18. Beispielabfragen zum Zugriff auf BioRS-Spalten vom Datentyp 'Date'*

Abfrage	Ergebnis
select e.entrydate from embl e where biors.CONTAINS(e.entrydate, date('11/01/1997')) = 1 fetch first 3 rows only	ENTRYDATE ----- 01-NOV-1997 01-NOV-1997 01-NOV-1997  3 Satz/Sätze ausgewählt.

Tabelle 18. Beispielabfragen zum Zugriff auf BioRS-Spalten vom Datentyp 'Date' (Forts.)

Abfrage	Ergebnis
select g.update from gen g where biors.CONTAINS(g.update, date('11/01/1997')) = 1 fetch first 3 rows only	UPDATE ----- 01-NOV-1997 11- 01-NOV-1997 12- 01-NOV-1997 06-
	3 Satz/Sätze ausgewählt.

Die Abfragen und Ergebnisse in Tabelle 19 zeigen die Verwendung der Funktionen BIOR.SCONTAINS\_LE und BIOR.SCONTAINS\_GE.

Tabelle 19. Beispielabfragen mit den Funktionen BIOR.SCONTAINS\_LE und BIOR.SCONTAINS\_GE

Abfrage	Ergebnis
select s.molweight from Swiss s where biors.CONTAINS_LE(s.molweight, 100368) = 1 fetch first 3 rows only	MOLWEIGHT ----- 100368 10576 8523
	3 Satz/Sätze ausgewählt.
select s.molweight from Swiss s where biors.CONTAINS_GE(s.molweight, 100368) = 1 fetch first 3 rows only	MOLWEIGHT ----- 100368 103625 132801
	3 Satz/Sätze ausgewählt.
select s.journalissue from Swiss s where biors.CONTAINS_GE(s.journalissue, 172) = 1 fetch first 3 rows only	JOURNALISSUE ----- 172 21 242 196
	3 Satz/Sätze ausgewählt.

Die Abfragen und Ergebnisse in Tabelle 20 auf Seite 97 zeigen, wie mit Hilfe der Funktion BIOR.SSEARCH\_TERM ein Suchbegriff im BioRS-Format eingegeben werden kann.

Tabelle 20. Beispielabfragen mit der Funktion BIORS.SEARCH\_TERM

Abfrage	Ergebnis
<pre>select s.publicationyear from Swiss s where biors.SEARCH_TERM (s.id, '[PublicationYear EQ number:1997;]')=1 fetch first 10 rows only</pre>	PUBLICATIONYEAR ----- 1997 1997 2000 1988 1991 1997 1994 1997 1997 1998 1994 1995 1997 1997 1999 1997 1994 1994 1995 1993 1992 1997  10 Satz/Sätze ausgewählt.
<pre>select s.molweight from Swiss s where biors.SEARCH_TERM (s.id, '[MolWeight EQ number:100368;]') = 1 fetch first 10 rows only</pre>	MOLWEIGHT ----- 100368 100368  2 Satz/Sätze ausgewählt.
<pre>select s.molweight from Swiss s where biors.SEARCH_TERM (s.id, '[MolWeight GREATER number:100368;]') = 1 fetch first 10 rows only</pre>	MOLWEIGHT ----- 100368 103625 132801 194328 130277 287022 289130 135502 112715 112599  10 Satz/Sätze ausgewählt.

Die folgende Abfrage zeigt, wie relationale Vergleichselemente verwendet werden, um eine Verknüpfung mit Gleichheitsattribut zwischen zwei Datenbanken zu erstellen, die in einer Elter-Kind-Abhängigkeit zueinander stehen:

```
select s.id, f.id, f.parent from Swiss s, Swissft f
where (f.parent = s.id) fetch first 10 rows only
```

Die Abfrageergebnisse sehen wie folgt aus:

ID	ID	PARENT
100K_RAT	100K_RAT.1	swissprot:100K_RAT
100K_RAT	100K_RAT.2	swissprot:100K_RAT
100K_RAT	100K_RAT.3	swissprot:100K_RAT
100K_RAT	100K_RAT.4	swissprot:100K_RAT
100K_RAT	100K_RAT.5	swissprot:100K_RAT
100K_RAT	100K_RAT.6	swissprot:100K_RAT
100K_RAT	100K_RAT.7	swissprot:100K_RAT
100K_RAT	100K_RAT.8	swissprot:100K_RAT
100K_RAT	100K_RAT.9	swissprot:100K_RAT
104K_THEPA	104K_THEPA.1	swissprot:104K_THEPA

10 Satz/Sätze ausgewählt.

In den vorstehenden Abfrageergebnissen ist der Eintrag 100K\_RAT das übergeordnete Element (Elter) für neun untergeordnete Elemente, den Kindern (100K\_RAT.1 bis 100K\_RAT.9).

**Zugehörige Konzepte:**

- „Richtlinien zur Leistungsoptimierung des BioRS-Wrappers“ auf Seite 98
- „Vergleichselemente für Verknüpfungen mit Gleichheitsattributen für den BioRS-Wrapper“ auf Seite 90

**Zugehörige Referenzen:**

- „Angepasste Funktionen und BioRS-Abfragen“ auf Seite 88
- „Anweisung CREATE NICKNAME - Beispiele für den BioRS-Wrapper“ auf Seite 86
- „Anweisungssyntax für CREATE NICKNAME - BioRS-Wrapper“ auf Seite 601

---

## Optimieren der Leistung des BioRS-Wrappers

### Richtlinien zur Leistungsoptimierung des BioRS-Wrappers

Dieser Abschnitt enthält Richtlinien dazu, wie die Abfrageleistung bei Verwendung des BioRS-Wrappers optimiert werden kann.

**Minimierung der zwischen Suchmaschinen übertragenen Datenmenge.**

Die Umgebung zusammenschlossener Datenbanken verwendet zwei Steuerkomponenten für die Abfrage. Für den BioRS-Wrapper sind diese Steuerkomponenten 'DB2<sup>®</sup> Universal Database' und 'BioRS'. Die DB2-Steuerkomponente verarbeitet Vergleichselemente (Vergleichsoperatoren wie beispielsweise =, BETWEEN, LIKE und <>), die für Kurznamenspalten angegeben sind. Die BioRS-Steuerkomponente verarbeitet Vergleichselemente, die mit Hilfe der vier angepassten Funktionen für den BioRS-Wrapper angegeben werden.

Um die Datenmenge, die zwischen den beiden Suchmaschinen übertragen wird, zu minimieren, strukturieren Sie Ihre Abfragen so, dass die Datenverarbeitung so oft wie möglich im Pushdown-Modus an das BioRS-System gesendet wird.

Wenn Sie in einer Abfrage Verknüpfungsoperationen durchführen müssen, nutzen Sie die in den BioRS-Datenbanken bereits vorhandenen Elter-Kind-Abhängigkeiten und führen Sie so oft wie möglich Verknüpfungsoperationen mit Gleichheitsattributen durch. Verknüpfungsoperationen mit Gleichheitsattributen werden in BioRS verarbeitet, was ebenfalls die Datenmenge minimiert, die zwischen den DB2- und BioRS-Steuerkomponenten für die Abfrage übertragen wird.

**Achtung:** Sie dürfen DB2 Information Integrator-Abfragen an BioRS nicht unterbrechen, indem Sie beispielsweise **Strg-D** oder **Strg-Z** im Befehlszeilenprozessor eingeben bzw. ein Anwendungsprogramm stoppen. Abgebrochene Abfragen hinterlassen so genannte "tote" Prozesse, die auf dem BioRS-Server ausgeführt werden. Diese "toten" Prozesse führen rasch zu einer Beeinträchtigung der Systemleistung sowohl von BioRS als auch von DB2 Information Integrator. Sind viele dieser "toten" Prozesse aktiv, können bei der Verarbeitung von Abfragen mit DB2 Information Integrator unerwartete Fehler auftreten. Eine gültige Abfrage beispielsweise könnte 0 Zeilen zurückgeben, obwohl Zeilen erwartet werden. In extremen Situatio-

nen kann es passieren, dass BioRS, DB2 Information Integrator oder beide Produkte gestoppt oder abnormal beendet werden.

### **Verwaltung von statistischen BioRS-Daten in der Umgebung zusammenschlossener Datenbanken.**

In einem System zusammenschlossener Datenbanken verwendet die zusammenschlossene Datenbank Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Das Verwalten von aktuellen Statistiken über die BioRS-Datenquellen ist von wesentlicher Bedeutung, um die Leistung des BioRS-Wrappers zu optimieren. Wenn sich die statistischen Daten oder Strukturmerkmale eines fernen Objekts, für das ein Kurzname definiert wurde, geändert haben, müssen Sie die Kardinalitätsstatistiken der entsprechenden Kurznamenspalte in Ihrem System zusammenschlossener Datenbanken aktualisieren.

Um die Leistung des BioRS-Wrappers zu optimieren, sollten Sie diese Aktualisierungen in regelmäßigen Abständen in DB2 Information Integrator durchführen.

#### **Zugehörige Konzepte:**

- „Optimieren der Abfrageverarbeitung“ in *Systeme zusammenschlossener Datenbanken*
- „Vergleichselemente für Verknüpfungen mit Gleichheitsattributen für den BioRS-Wrapper“ auf Seite 90
- „Statistische BioRS-Daten“ auf Seite 99

#### **Zugehörige Referenzen:**

- „Angepasste Funktionen und BioRS-Abfragen“ auf Seite 88
- „BioRS-Wrapper - Beispielabfragen“ auf Seite 93

## **Statistische BioRS-Daten**

In einem System zusammenschlossener Datenbanken verwendet die zusammenschlossene Datenbank Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Diese Statistiken werden aus BioRS-Datenquellen abgerufen, wenn Sie unter Verwendung der Anweisung CREATE NICKNAME einen Kurznamen erstellen. Die zusammenschlossene Datenbank stellt das Vorhandensein des Objekts in der Datenquelle sicher und versucht dann, statistische Daten zu bestehenden Datenquellen zu sammeln. Informationen werden aus den Datenquellenkatalogen gelesen und in den DB2<sup>®</sup>-Katalog des Systems zusammenschlossener Datenbanken auf dem Server mit zusammenschlossenen Datenbanken aufgenommen.

Für BioRS-Datenquellen sind folgende statistische Daten von wesentlicher Bedeutung:

- Die Kardinalität eines Kurznamens. Bei BioRS-Datenquellen entspricht die Kardinalität eines Kurznamens der Anzahl von Einträgen in der zugehörigen BioRS-Datenbank.
- Die Kardinalität der Spalte, die dem Element BioRS\_ID\_ entspricht. Die Kardinalität dieser Spalte muss der Kardinalität des Kurznamens entsprechen, in dem auf die Spalte verwiesen wird.
- Die Kardinalität aller Spalten, die der BioRS-Wrapper möglicherweise verwenden muss.



Um die Leistung des BioRS-Wrappers zu optimieren, müssen aktuelle Statistiken über die BioRS-Datenquellen verwaltet werden. Wenn sich die statistischen Daten oder Strukturmerkmale eines fernen Objekts, für das ein Kurzname definiert wurde, ändern, müssen Sie die entsprechenden Kardinalitätsstatistiken in Ihrem System zusammengeschlossener Datenbanken aktualisieren. Die Kardinalitätsstatistiken werden in der Katalogsicht SYSSTAT.TABLES und in der Katalogsicht SYSSTAT.COLUMNS gespeichert.

Um die BioRS-Kardinalitätsstatistiken in Ihrem System zusammengeschlossener Datenbanken zu verwalten, führen Sie die folgenden Tasks durch:

1. Ermitteln Sie ggf. die Kardinalitätsstatistiken des gewünschten Kurznamens.
2. Aktualisieren Sie die entsprechenden Kardinalitätsstatistiken in den erforderlichen Katalogsichten.

#### **Zugehörige Konzepte:**

- „Optimieren der Abfrageverarbeitung“ in *Systeme zusammengeschlossener Datenbanken*

#### **Zugehörige Tasks:**

- „Ermitteln der Kardinalitätsstatistiken von BioRS-Datenbanken“ auf Seite 100
- „Aktualisieren der Kardinalitätsstatistiken von BioRS-Kurznamen“ auf Seite 100
- „Aktualisieren der Kardinalitätsstatistiken von BioRS-Spalten“ auf Seite 101
- „Aktualisieren der Kardinalität der BioRS-Spalte `_ID_`“ auf Seite 103

## **Ermitteln der Kardinalitätsstatistiken von BioRS-Datenbanken**

Bevor Sie Kurznamenstatistiken oder die Kardinalität der dem BioRS-Element `_ID_` entsprechenden Spalte aktualisieren können, müssen Sie zunächst die Kardinalitätsstatistiken der entsprechenden BioRS-Datenbanken ermitteln.

#### **Vorgehensweise:**

Um die Kardinalität einer bestimmten Datenbank in BioRS zu ermitteln, verwenden Sie das BioRS-Dienstprogramm `'admin_find'` oder `'www_find.cgi'`. Geben Sie die Option `-c` (Kardinalität) an. Die BioRS-Dokumentation enthält weitere Informationen zu diesen beiden BioRS-Dienstprogrammen.

#### **Zugehörige Konzepte:**

- „Statistische BioRS-Daten“ auf Seite 99

#### **Zugehörige Tasks:**

- „Aktualisieren der Kardinalitätsstatistiken von BioRS-Kurznamen“ auf Seite 100
- „Aktualisieren der Kardinalitätsstatistiken von BioRS-Spalten“ auf Seite 101
- „Aktualisieren der Kardinalität der BioRS-Spalte `_ID_`“ auf Seite 103

## **Aktualisieren der Kardinalitätsstatistiken von BioRS-Kurznamen**

Sie müssen die Kardinalitätsstatistiken von BioRS-Kurznamen aktualisieren, wenn sich der Inhalt einer BioRS-Datenbank, für die Sie einen Kurznamen erstellt haben, maßgeblich ändert. Durch ordnungsgemäße Verwaltung der Kardinalitätsstatistiken

von Kurznamen können das Optimierungsprogramm und der BioRS-Wrapper den Datenzugriffsplan auswählen, der die beste Leistung erbringt.

Um die Kardinalitätsstatistiken von BioRS-Kurznamen zu aktualisieren, modifizieren Sie die Katalogsicht SYSSTAT.TABLES anhand der korrekten Kardinalitätszahl.

#### **Voraussetzungen:**

Sie müssen die Kardinalitätszahl der BioRS-Datenbank ermitteln, die dem Kurznamen entspricht, dessen Statistiken Sie aktualisieren wollen.

#### **Vorgehensweise:**

Setzen Sie die Anweisung UPDATE mit folgender Syntax ab:

```
UPDATE sysstat.tables SET card=kardinalität
WHERE tabschema=kurznamenschema
AND tablename=name_des_kurznamens;
```

- *kardinalität* ist die Kardinalitätszahl der BioRS-Datenbank, die dem Kurznamen mit den zu aktualisierenden Statistikdaten entspricht.
- *kurznamenschema* ist der Name des Schemas, das dem Kurznamen mit den zu aktualisierenden Statistikdaten zugeordnet ist.
- *name\_des\_kurznamens* ist der Name des Kurznamens, dessen Statistikdaten Sie aktualisieren möchten.

#### **Zugehörige Konzepte:**

- „Statistische BioRS-Daten“ auf Seite 99

#### **Zugehörige Tasks:**

- „Ermitteln der Kardinalitätsstatistiken von BioRS-Datenbanken“ auf Seite 100
- „Aktualisieren der Kardinalitätsstatistiken von BioRS-Spalten“ auf Seite 101
- „Aktualisieren der Kardinalität der BioRS-Spalte `_ID_`“ auf Seite 103

## **Aktualisieren der Kardinalitätsstatistiken von BioRS-Spalten**

Um die Kardinalitätsstatistiken von BioRS-Spalten im System zusammengeschlüssener Datenbanken zu aktualisieren, müssen Sie die Katalogsicht SYSSTAT.COLUMNS modifizieren.

Durch ordnungsgemäße Verwaltung der Kardinalitätsstatistiken von BioRS-Spalten können das Optimierungsprogramm und der BioRS-Wrapper während der Abfrageverarbeitung denjenigen Datenzugriffsplan auswählen, der die beste Leistung erbringt.

Sie haben die Möglichkeit, die Kardinalitätsstatistiken von BioRS-Spalten als Teil der umfangreicheren Task des Hinzufügens von BioRS zu einem System zusammengeschlüssener Datenbanken zu aktualisieren. Außerdem können Sie die Kardinalitätsstatistiken von BioRS-Spalten auch dann aktualisieren, wenn Sie die Abfrageleistung für BioRS-Datenquellen verbessern wollen.

#### **Einschränkungen:**

Sie dürfen diese Vorgehensweise nicht verwenden, um die Kardinalitätsstatistiken für Spalten zu ändern, die dem BioRS-Element `_ID_` entsprechen. Um die Kardina-

litätsstatistiken von Spalten zu ändern, die dem BioRS-Element `_ID_` entsprechen, ist eine andere Vorgehensweise erforderlich.

### Vorgehensweise:

Um die Kardinalitätsstatistiken von BioRS-Spalten zu aktualisieren, setzen Sie die Anweisung UPDATE mit folgender Syntax ab:

```
UPDATE sysstat.columns SET colcard=(SELECT COUNT(DISTINCT spaltenname)
  FROM kurznamen.schema.name_des_kurznamens)
WHERE
  tabschema=kurznamen.schema
  AND tabname=name_des_kurznamens
  AND colname=spaltenname;
```

- *spaltenname* ist der Name der Spalte, deren Kardinalitätsstatistik Sie aktualisieren möchten.
- *kurznamen*.*schema* ist der Name des Schemas, das dem Kurznamen zugeordnet ist, in dem die angegebene Spalte verwendet wird.
- *name\_des\_kurznamens* ist der Name des Kurznamens, in dem die angegebene Spalte verwendet wird.

Die Ausführung der Abfrage kann einige Minuten in Anspruch nehmen, da alle Einträge für die im Kurznamen angegebene Datenbank abgerufen werden müssen.

Wenn eine Spalte mehrere Werte enthalten kann (beispielsweise beim Format des Elements 'PublicationYear' der Datenbank 'SwissProt'), wird die Berechnung für die Verwendung von SQL-Abfragen zu komplex. Für solche Spalten müssen Sie den Kardinalitätswert manuell berechnen und die Katalogsicht SYSSTAT.COLUMNS anschließend entsprechend aktualisieren. Um den Kardinalitätswert zu berechnen, dividieren Sie die Anzahl der unterschiedlichen Werte in der Spalte durch die durchschnittliche Anzahl an Werten pro Zeile. Der berechnete Kardinalitätswert kann nicht größer sein als die Kardinalität der Tabelle.

### Beispiel:

Angenommen, Sie haben einen Kurznamen mit drei Zeilen. Die Werte der Spalte 'PublicationYear' für diese drei Zeilen lauten wie folgt:

- 1997 1992 1985
- 1997 1992 1982
- 1992 1991 1990 1976 1974 1971

Es gibt neun unterschiedliche Werte, und die durchschnittliche Anzahl an Werten in einer Zeile beträgt vier. Die Kardinalität dieser Spalte 'PublicationYear' ist demnach  $9:4$  bzw.  $3$  (2,25 aufgerundet auf die nächsthöhere ganze Zahl). Nachdem Sie nun die Kardinalität berechnet haben, können Sie die Katalogsicht SYSSTAT.COLUMNS mit Hilfe der folgenden Anweisung UPDATE entsprechend aktualisieren:

```
UPDATE sysstat.columns SET colcard=3
WHERE
  tabschema=kurznamen.schema
  AND tabname=name_des_kurznamens
  AND colname=spaltenname
```

- $3$  ist der Kardinalitätswert der Spalte.
- *kurznamen*.*schema* ist der Name des Schemas, das dem zu Grunde liegenden Kurznamen zugeordnet ist, in dem die angegebene Spalte verwendet wird.
- *name\_des\_kurznamens* ist der Name des zu Grunde liegenden Kurznamens, in dem die angegebene Spalte verwendet wird.

- *spaltenname* ist der Name der Spalte, deren Kardinalitätsstatistik Sie aktualisieren möchten.

**Zugehörige Konzepte:**

- „Statistische BioRS-Daten“ auf Seite 99

**Zugehörige Tasks:**

- „Aktualisieren der Kardinalitätsstatistiken von BioRS-Kurznamen“ auf Seite 100
- „Aktualisieren der Kardinalität der BioRS-Spalte *\_ID\_*“ auf Seite 103

## Aktualisieren der Kardinalität der BioRS-Spalte *\_ID\_*

Durch ordnungsgemäße Verwaltung der Kardinalitätsstatistiken der Spalte, die dem BioRS-Element *\_ID\_* zugeordnet ist, können das Optimierungsprogramm und der BioRS-Wrapper den Datenzugriffsplan auswählen, der die beste Leistung erbringt. Um die Kardinalitätszahl der Spalte zu aktualisieren, die dem BioRS-Element *\_ID\_* zugeordnet ist, müssen Sie die Katalogsicht SYSSTAT.COLUMNS modifizieren.

**Voraussetzungen:**

Sie müssen die Kardinalitätszahl der BioRS-Datenbank ermitteln, die dem Kurznamen entspricht, in dem auf die Spalte verwiesen wird. Die Kardinalitätszahl der Spalte, die dem BioRS-Element *\_ID\_* zugeordnet ist, muss der Kardinalität des Kurznamens entsprechen, in dem auf die Spalte verwiesen wird.

**Vorgehensweise:**

Um die Kardinalitätsstatistiken der BioRS-Spalte *\_ID\_* zu aktualisieren, setzen Sie die Anweisung UPDATE mit folgender Syntax ab:

```
UPDATE sysstat.columns SET colcard=<kardinalität>
WHERE
    tabschema=<kurznamenschema>
    AND tabname=<name_des_kurznamens>
    AND colname IN (SELECT colname FROM syscat.coloptions
WHERE
    tabschema=<name_des_kurznamens>
    AND tabname=<name_des_kurznamens>
    AND option='ELEMENT_NAME';
    AND setting='_ID_')
```

- *kardinalität* ist die Kardinalitätszahl der BioRS-Datenbank, die dem Kurznamen der Spalte entspricht.
- *kurznamenschema* ist der Name des Schemas, das dem Kurznamen der Spalte zugeordnet ist.
- *name\_des\_kurznamens* ist der Name der Kurzname, in dem die Spalte verwendet wird.

**Zugehörige Konzepte:**

- „Statistische BioRS-Daten“ auf Seite 99

**Zugehörige Tasks:**

- „Ermitteln der Kardinalitätsstatistiken von BioRS-Datenbanken“ auf Seite 100
- „Aktualisieren der Kardinalitätsstatistiken von BioRS-Kurznamen“ auf Seite 100
- „Aktualisieren der Kardinalitätsstatistiken von BioRS-Spalten“ auf Seite 101

## Nachrichten für den BioRS-Wrapper

In diesem Abschnitt werden die Nachrichten erläutert, die bei der Verwendung des BioRS-Wrappers angezeigt werden können.

Tabelle 21. Vom BioRS-Wrapper ausgegebene Nachrichten

Fehlercode	Nachricht	Erläuterung
SQL0604N	Das Attribut 'length', 'precision' oder 'scale' für die Spalte, den einzigartigen Datentyp, den strukturierten Typ, das Attribut eines strukturierten Typs, die Funktion oder Typzuordnung <datenelement> ist nicht gültig.	Der Datentyp für eine Kurznamenspalte ist mit dem BioRS-Typ des zugrunde liegenden Datenbankelements nicht kompatibel. Überprüfen Sie den Datentyp der Spalte in der Anweisung CREATE NICKNAME.
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Fehler beim Erstellen eines Wrapperobjekts.")	Beim Erstellen eines neuen Wrapperobjekts ist ein Fehler aufgetreten. Wenden Sie sich an die IBM Unterstützungsfunktion.
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "BioRS <tracepunkt>/<code>.")	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Unterstützungsfunktion.
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Zuordnung von Speicherkapazität ist fehlgeschlagen: <tracepunkt>.")	Bei der Speicherzuordnung trat ein Fehler auf. Stellen Sie sicher, dass für den Host des Servers mit zusammengeschlossenen Datenbanken ausreichend Speicherkapazität zur Verfügung steht und übergeben Sie die Abfrage erneut. Falls der Fehler weiterhin auftritt, wenden Sie sich an die IBM Unterstützungsfunktion.
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "sqlno_crule_save_plans[100]:rc(-214272209) Leere Planliste.")	Das Optimierungsprogramm und der BioRS-Wrapper konnten keinen gemeinsamen Plan für die Ausführung der Abfrage finden. Vereinfachen Sie die Abfrage und führen Sie sie erneut aus.
SQL0401N	Die Datentypen der Operanden für die Operation "=" sind nicht kompatibel.	Die Abfrage ist ungültig, da der Ausdruck auf der rechten Seite im Vergleichselement einer angepassten Funktion eine ganze Zahl sein muss.
SQL1822N	Unerwarteter Fehlercode "" von Datenquelle "BioRS-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Datenbank wurde nicht gefunden".	Die BioRS-Datenbank, auf die in der Anweisung CREATE NICKNAME verwiesen wurde, konnte auf dem BioRS-Server nicht gefunden werden. Überprüfen Sie die Anweisung CREATE NICKNAME und stellen Sie sicher, dass der Name der Datenbank korrekt ist.

Tabelle 21. Vom BioRS-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Unerwarteter Fehlercode "" von Datenquelle "BioRS-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Zeitsperre für die Verbindung wurde erreicht".	Der BioRS-Server hat auf eine Kommunikationsanforderung nicht innerhalb des von der Option TIMEOUT angegebenen Zeitraums geantwortet.
SQL1822N	Unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "BioRS-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Lesen vom Server."	Während des Lesens von Daten vom BioRS-Server trat ein Kommunikationsfehler auf. Der Wert des Fehlercodes <tracepunkt> enthält möglicherweise weitere Informationen zu diesem Fehler.
SQL1822N	Unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "BioRS-Wrapper" empfangen. Zugeordneter Text und Token sind "Der Host wurde nicht gefunden".	Der in der Serveroption HOST angegebene Host des BioRS-Servers wurde nicht gefunden. Überprüfen Sie die Anweisung CREATE SERVER und stellen Sie sicher, dass der Wert der Serveroption HOST korrekt ist.
SQL1822N	Unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "BioRS-Wrapper" empfangen. Zugeordneter Text und Token sind "Verbindung zum Server kann nicht hergestellt werden".	Der Wrapper konnte keine Verbindung zu dem Server herstellen, der von der Serveroption HOST angegeben wurde. Der Wert des Fehlercodes <tracepunkt> enthält möglicherweise weitere Informationen zu diesem Fehler.
SQL1822N	Unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "BioRS-Wrapper" empfangen. Zugeordneter Text und Token sind "TCPIP-Socket konnte nicht erstellt werden".	Der Wrapper konnte keinen TCP/IP-Socket erstellen. Der Wert des Fehlercodes <tracepunkt> enthält möglicherweise weitere Informationen zu diesem Fehler.
SQL1822N	Unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "BioRS-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Senden zum Server."	Der Wrapper konnte eine Anforderung nicht an den BioRS-Server senden. Der Wert des Fehlercodes <tracepunkt> enthält möglicherweise weitere Informationen zu diesem Fehler.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Groß-/Kleinschreibung des Servers kann nicht geändert werden."	Der Wert der Serveroption CASE_SENSITIVE kann nicht mit Hilfe von SQL-Anweisungen geändert werden. Um den Wert dieser Option zu ändern, müssen Sie den Server zunächst löschen. Anschließend müssen Sie den Server mit Hilfe der Anweisung CREATE SERVER erneut erstellen und dann den korrekten Wert für die Option CASE_SENSITIVE angeben.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Mehrere Verknüpfungen zwischen zwei Kurznamen."	Die Abfrage ist ungültig, da zwischen zwei Kurznamen jeweils nur ein Verknüpfungsprädikat zulässig ist.



Tabelle 21. Vom BioRS-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Die rechte Seite eines Funktionsprädikats muss eine Konstante sein."	Die Abfrage ist ungültig, da der Ausdruck auf der rechten Seite des Prädikats einer angepassten Funktion eine Konstante sein muss.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Argument 1 der angepassten Funktion ist keine Spalte."	Die Abfrage ist ungültig, da das erste Argument einer angepassten Funktion auf eine Spalte eines BioRS-Kurznamens verweisen muss.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Argument 1 der Funktion CONTAINS ist nicht indexiert."	Die Abfrage ist ungültig. Die Spalte, auf die im ersten Argument der Funktion BIOR.S.CONTAINS, BIOR.S.CONTAINS_LE oder BIOR.S.CONTAINS_GE verwiesen wird, muss eine indexierte Spalte sein.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Ungültiger Typ für Argument 1 der Funktion <funktionsname>."	Die Abfrage ist ungültig. Der Datentyp der Spalte, auf die im ersten Argument der Funktion BIOR.S.CONTAINS, BIOR.S.CONTAINS_LE oder BIOR.S.CONTAINS_GE verwiesen wird, ist nicht korrekt.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Argument 1 der Funktion SEARCH_TERM ist nicht die Spalte _ID_."	Die Abfrage ist ungültig. Der Spalte, auf die im ersten Argument der Funktion SEARCH_TERM verwiesen wird, ist nicht dem BioRS-Element _ID_ zugeordnet.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Der Bindeparameter kann nicht NULL sein."	Der Wert einer Spalte oder Hostvariablen, auf den im zweiten Argument der Funktion BIOR.S.CONTAINS verwiesen wurde, war NULL. Der BioRS-Wrapper kann keine Nullwerte verarbeiten.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Wert kann nicht in BioRS-Literal umgewandelt werden."	In einem Literal, einer Spalte oder einer Hostvariable wurde ein Wert an den Wrapper übergeben, der nicht in ein gültiges BioRS-Literal umgewandelt werden konnte.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Die Serverversion kann nicht geändert werden."	Die Serverversion kann nicht mit Hilfe der Anweisung ALTER SERVER geändert werden. Um die Serverversion zu ändern, müssen Sie den Server zunächst löschen. Anschließend müssen Sie den Server mit Hilfe der Anweisung CREATE SERVER erneut erstellen und dabei die korrekte Version angeben.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Ungültiger Typ für Argument 2 der Funktion <funktionsname>."	Die Abfrage ist ungültig. Der Datentyp der Spalte, auf die im zweiten Argument der Funktion BIOR.S.CONTAINS, BIOR.S.CONTAINS_LE oder BIOR.S.CONTAINS_GE verwiesen wird, ist nicht korrekt.

Tabelle 21. Vom BioRS-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Der Kurzname hat keine Spalten."	Für die Anweisung CREATE NICKNAME wurden keine Spaltendeklarationen angegeben. Für das Erstellen von Kurznamen sind Spaltendeklarationen jedoch erforderlich.

**Zugehörige Konzepte:**

- „Nachrichten - Einführung“ in *Fehlernachrichten Band 1*

**Zugehörige Referenzen:**

- „SQLSTATE-Nachrichten“ in *Fehlernachrichten Band 2*





---

## Kapitel 7. Konfigurieren des Zugriffs auf BLAST-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in BLAST-Datenquellen gespeichert sind. Der Zugriff auf BLAST-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält folgende Informationen:

- Erläuterungen zu BLAST.
- Liste der auszuführenden Tasks.
- Beispiele zu den benötigten SQL-Anweisungen.
- Liste der dem BLAST-Wrapper zugeordneten Fehlermeldungen.

---

### Was ist BLAST?

BLAST (Basic Local Alignment Search Tool) ist ein Dienstprogramm, das von NCBI (National Center for Biotechnology Information) verwaltet wird. BLAST wird verwendet, um eine Datenbank mit Nukleotid- oder Aminosäuresequenzen auf "Übereinstimmungen" zu durchsuchen. Eine BLAST-Übereinstimmung enthält ein oder mehrere HSPs (High-Scoring Segment Pairs). Bei einem HSP handelt es sich um ein Paar von Sequenzfragmenten mit maximal lokalem Alignment und einem Ähnlichkeitsscore, der einen bestimmten Schwellenwert übersteigt. NCBI stellt eine ausführbare Datei (blastall) zur Verfügung, mit der BLAST-Suchen in BLAST-fähigen Datenquellen wie beispielsweise GenBank und SWISS-PROT ausgeführt werden können.

Der BLAST-Wrapper unterstützt alle fünf BLAST-Suchtypen: BLASTn, BLASTp, BLASTx, tBLASTn und tBLASTx. Diese Suchtypen werden in Tabelle 22 beschrieben.

*Tabelle 22. Vom BLAST-Wrapper unterstützte BLAST-Suchtypen*

BLAST-Suchtyp	Beschreibung
BLASTn	Ein BLAST-Suchtyp, bei dem eine Nukleotidsequenz mit dem Inhalt einer Nukleotidsequenzdatenbank verglichen wird, um Sequenzen mit Bereichen zu ermitteln, die mit Bereichen der ursprünglichen Sequenz übereinstimmen, d. h. homolog sind.
BLASTp	Ein BLAST-Suchtyp, bei dem eine Aminosäuresequenz mit dem Inhalt einer Aminosäuresequenzdatenbank verglichen wird, um Sequenzen mit Bereichen zu ermitteln, die mit Bereichen der ursprünglichen Sequenz übereinstimmen, d. h. homolog sind.
BLASTx	Ein BLAST-Suchtyp, bei dem eine Nukleotidsequenz mit dem Inhalt einer Aminosäuresequenzdatenbank verglichen wird, um Sequenzen mit Bereichen zu ermitteln, die mit Bereichen der ursprünglichen Sequenz übereinstimmen, d. h. homolog sind. Die Abfragesequenz wird in allen sechs Leserahmen umgesetzt, und jede der Ergebnissequenzen wird für die Suche in der Sequenzdatenbank verwendet.

Tabelle 22. Vom BLAST-Wrapper unterstützte BLAST-Suchtypen (Forts.)

BLAST-Suchtyp	Beschreibung
tBLASTn	Ein BLAST-Suchtyp, bei dem eine Aminosäuresequenz mit dem Inhalt einer Nukleotidsequenzdatenbank verglichen wird, um Sequenzen mit Bereichen zu ermitteln, die mit Bereichen der ursprünglichen Sequenz übereinstimmen, d. h. homolog sind. Die Sequenzen in der Sequenzdatenbank werden in allen sechs Leserahmen umgesetzt, und die Ergebnissequenz werden auf Bereiche hin durchsucht, die mit Bereichen der Abfragesequenz übereinstimmen, d. h. homolog sind.
tBLASTx	Ein BLAST-Suchtyp, bei dem eine Nukleotidsequenz mit dem Inhalt einer Nukleotidsequenzdatenbank verglichen wird, um Sequenzen mit Bereichen zu ermitteln, die mit Bereichen der ursprünglichen Sequenz übereinstimmen, d. h. homolog sind. Bei einer tBLASTx-Suche werden sowohl die Abfragesequenz als auch die Sequenzdatenbank in allen sechs Leserahmen umgesetzt, und die Ergebnissequenzen werden miteinander verglichen, um homologe Bereiche zu ermitteln.

Abb. 7 zeigt die Funktionsweise von BLAST in Ihrem System zusammengeschlossener Datenbanken.

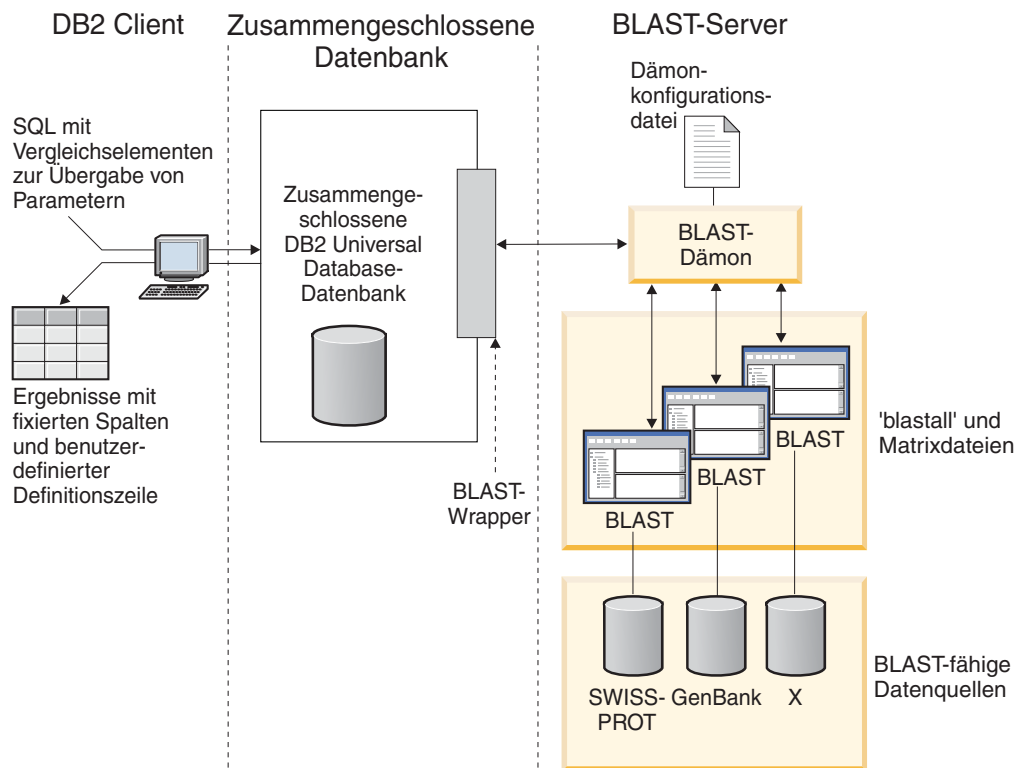


Abbildung 7. Funktionsweise des BLAST-Wrappers

Auf der Clientseite übergeben Benutzer oder Anwendungen SQL-Anweisungen mit BLAST-spezifischen Vergleichselementen (Prädikaten), die Parameter übergeben und standardmäßigen BLAST-Optionen zugeordnet werden. Die SQL-Anweisungen mit den Eingabevergleichselementen werden an das entsprechende DB2<sup>®</sup> Universal Database-System zusammenschlossener Datenbanken gesendet, auf dem der BLAST-Wrapper installiert ist.

Der BLAST-Wrapper setzt die Abfrage in ein Format um, das von der BLAST-Anwendung verarbeitet werden kann, und sendet die umgesetzte Abfrage an den entsprechenden BLAST-Server. Bei diesem Server kann es sich um eine Maschine handeln, die von der Maschine mit dem System zusammenschlossener Datenbanken getrennt ist. Auf dem BLAST-Server wird ein spezielles Dämonprogramm ausgeführt. Dieser Dämon, der Informationen von einer entsprechenden Dämonkonfigurationsdatei verwendet, empfängt die Abfrageanforderung vom System zusammenschlossener Datenbanken und sendet diese Anforderung an die BLAST-Anwendung. Anschließend wird die BLAST-Anwendung auf die übliche Weise für eine BLAST-fähige Datenquelle ausgeführt.

Die Ergebnisse werden zunächst an BLAST und anschließend an den Dämon zurückgegeben. Der Dämon wiederum gibt die abgerufenen Daten an den BLAST-Wrapper zurück. Der Wrapper setzt die Daten nun in ein relationales Tabellenformat um und gibt diese Tabelle an den Benutzer bzw. die Anwendung zurück. Die zurückgegebenen Daten setzen sich aus zwei Teilen zusammen:

- aus einer Reihe von standardmäßigen, fixierten Spalten, die den BLAST-Benutzern bekannt sind, und
- aus von Benutzern konfigurierten Definitionszeileninformationen.

Das folgende Beispiel zeigt, wie relationale Informationen aus den BLAST-fähigen Datenquellen extrahiert werden. Die Daten werden aus einem unformatierten FASTA-Dateiformat in eine BLAST-fähige Datei verschoben und weiter in eine relationale Tabelle, die mit anderen Datenquellen in Ihrem System zusammenschlossener Datenbanken verknüpft werden kann.

Abb. 8 auf Seite 112 zeigt eine FASTA-Beispieldatei mit vier Einträgen für Definitionszeilen und Nukleotidsequenzen.



Tabelle 23. BLAST gibt Ergebnisse in einem relationalen Tabellenformat zurück, sofern dies im entsprechenden System zusammengeschlossener Datenbanken integriert ist.

Unique ID	Experiment number	Organism number	HSP_INFO	SCORE
PMON5744	4986	7	Identities = 57/201 (28%), Positives = 57/201 (28%), Gaps = 0/201 (0%)	+1.134870000000000E+002
PMON5426	8924	15	Identities = 35/201 (17%), Positives = 35/201 (17%), Gaps = 0/201 (0%)	+6.987540000000000E+001
PMON5498	6512	13	Identities = 26/201 (13%), Positives = 26/201 (13%), Gaps = 0/201 (0%)	+5.203420000000000E+001

Die Daten liegen in einem vollständig relationalen Format vor und können mit Daten aus anderen von Ihrem Labor verwendeten Datenquellen verknüpft werden. Das Kombinieren der Ergebnisse aus mehreren Datenquellen kann zu Erkenntnissen führen, die vor der Implementierung des Systems zusammengeschlossener Datenbanken nicht so einfach oder effizient erlangt worden wären.

**Zugehörige Tasks:**

- „Hinzufügen von BLAST-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 113

---

## Hinzufügen von BLAST zu einem Server mit zusammengeschlossenen Datenbanken

### Hinzufügen von BLAST-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf BLAST-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken auf BLAST-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

**Voraussetzungen:**

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.

### Vorgehensweise:

Führen Sie folgende Schritte aus, um BLAST-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Vergewissern Sie sich, dass die korrekte Version der ausführbaren Datei 'blastall' und der Matrixdateien installiert wurde.
2. Konfigurieren Sie den BLAST-Dämon.
3. Starten Sie den BLAST-Dämon.
4. Registrieren Sie den Wrapper.
5. Registrieren Sie die Serverdefinition.
6. Registrieren Sie Kurznamen für BLAST-Suchen.

### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

### Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Überprüfen, ob die korrekte Version der ausführbaren Datei 'blastall' und der Matrixdateien installiert wurde“ auf Seite 114
- „Konfigurieren des BLAST-Dämons“ auf Seite 115
- „Starten des BLAST-Dämons“ auf Seite 119
- „Registrieren des BLAST-Wrappers“ auf Seite 121
- „Registrieren des Servers für BLAST-Datenquellen“ auf Seite 122
- „Registrieren von Kurznamen für BLAST-Datenquellen“ auf Seite 123
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43

### Zugehörige Referenzen:

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## Überprüfen, ob die korrekte Version der ausführbaren Datei 'blastall' und der Matrixdateien installiert wurde

Die Überprüfung, ob die korrekte Version der Datei 'blastall' und der Matrixdatei installiert wurde, ist Teil der umfangreicheren Task des Hinzufügens von BLAST zu einem System zusammengeschlossener Datenbanken.

### Voraussetzungen:

Vergewissern Sie sich, dass auf Ihrer BLAST-Servermaschine die jeweils neueste Version der ausführbaren Datei 'blastall' und der Matrixdateien BLOSUM62, BLOSUM80, PAM30 und PAM70 installiert ist. Verfügen Sie nicht über die neueste Version der ausführbaren Datei 'blastall', müssen Sie die Binär- und Matrixdateien installieren. Die Matrixdateien müssen sich in demselben Verzeichnis befinden wie die ausführbare Datei 'blastall'.

### Vorgehensweise:

Um den Versionsstand der ausführbaren Datei 'blastall' und der Matrixdateien zu überprüfen, gehen Sie wie folgt vor:

1. Führen Sie über die Befehlszeile eine BLAST-Suche aus und notieren Sie die Versionsnummer, die in der Ausgabedatei angegeben wird.
2. Überprüfen Sie die Produktwebsite auf mit diesem Wrapper getestete BLAST-Versionen, um sicherzustellen, dass Sie über eine unterstützte Version verfügen.

Die nächste Task in dieser Tasksequenz ist das Konfigurieren des BLAST-Dämons.

### Zugehörige Tasks:

- „Konfigurieren des BLAST-Dämons“ auf Seite 115

## Konfigurieren des BLAST-Dämons

Das Konfigurieren des BLAST-Dämons ist Teil der umfangreicheren Task des Hinzufügens von BLAST zu einem System zusammenschlossener Datenbanken.

Der BLAST-Wrapper erfordert einen BLAST-Dämon. Dieser BLAST-Dämon muss auf einem Server ausgeführt werden, auf den Sie von Ihrem System zusammenschlossener Datenbanken aus über TCP/IP zugreifen können. Dabei kann dies derselbe Server sein, der als Server mit zusammenschlossenen Datenbanken betrieben wird, oder es kann ein separater BLAST-Server sein.

Der Dämon wird getrennt vom Wrapper und der zusammenschlossenen Datenbank ausgeführt. Er ist empfangsbereit für BLAST-Jobanforderungen vom Wrapper.

### Voraussetzungen:

Der BLAST-Dämon muss über Folgendes verfügen:

- Ausführungszugriff auf die Binärdatei blastall, sodass BLAST-Suchen ausgeführt werden können.
- Schreibzugriff auf ein Verzeichnis, in das temporäre Dateien geschrieben werden können.
- Lesezugriff auf mindestens eine BLAST-fähige Datenquelle, für die BLAST-Suchen ausgeführt werden können. Die ausführbare Datei blastall muss über Lesezugriff für die Datendatei und die BLAST-Indexdateien verfügen, die vom Programm 'formatdb' generiert werden.

### Einschränkungen:

Der BLAST-Dämon wird unter Umständen nicht korrekt ausgeführt, wenn der Pfad der ausführbaren Datei oder der Datenbank Leerzeichen enthält. Sie sollten beispielsweise die ausführbare BLAST-Datei auf Windows-Servern nicht unter C:\Program Files installieren.

### Vorgehensweise:

Gehen Sie wie folgt vor, um den BLAST-Dämon zu konfigurieren:

1. Stellen Sie sicher, dass die ausführbaren Dateien des BLAST-Dämons sich auf dem korrekten Server befinden. Während der Installation von DB2 Information Integrator werden die ausführbaren Dämondateien in einem Verzeichnis auf dem Server mit zusammenschlossenen Datenbanken installiert:



#### Unter UNIX:

Die ausführbare Dämondatei ist `db2blast_daemon`. Diese Datei wird im Verzeichnis `$DB2PATH/bin` installiert.

#### Unter Windows:

Die ausführbaren Dämondateien sind `db2blast_daemon.exe` und `db2blast_daemon_svc.exe`. Diese Dateien werden im Verzeichnis `%DB2PATH%\bin` installiert.

Wenn Sie einen separaten Computer als BLAST-Server verwenden, müssen Sie die ausführbaren Dämondateien von dem Verzeichnis auf dem Server mit zusammengeschlossenen Datenbanken in ein Verzeichnis auf dem Computer kopieren, der als BLAST-Server dient. Die ausführbaren Dämondateien können in einem beliebigen Verzeichnis auf der BLAST-Servermaschine ausgeführt werden, wenn der Name im Verzeichnispfad keine Leerzeichen enthält.

2. Stellen Sie sicher, dass die Konfigurationsdatei des BLAST-Dämons sich auf dem korrekten Server befindet. Während der Installation von DB2 Information Integrator wird eine Beispiel-Dämonkonfigurationsdatei, `BLAST_DAEMON.config`, in einem Verzeichnis auf dem Server mit zusammengeschlossenen Datenbanken installiert:

#### Unter UNIX:

Die Dämonkonfigurationsdatei wird im Verzeichnis `$DB2PATH/bin` installiert. `$DB2PATH` ist das Verzeichnis, in dem DB2 Information Integrator installiert ist.

#### Unter Windows:

Die Dämonkonfigurationsdatei wird im Verzeichnis `%DB2PATH%\bin` installiert. `%DB2PATH%` ist das Verzeichnis, in dem DB2 Information Integrator installiert ist (normalerweise ist dies `C:\SQLLIB\bin`).

Der Dämon sucht die Konfigurationsdatei standardmäßig in dem Arbeitsverzeichnis, von dem aus der Dämon gestartet wird. Sie können die Konfigurationsdatei in eine andere Position kopieren. Wenn Sie eine BLAST-Servermaschine verwenden, müssen Sie die Dämonkonfigurationsdatei aus dem Verzeichnis auf dem Server mit zusammengeschlossenen Datenbanken in ein Verzeichnis auf der BLAST-Servermaschine kopieren. Sie können die Dämonkonfigurationsdatei in ein beliebiges Verzeichnis auf der BLAST-Servermaschine kopieren, auf das der Dämon zugreifen kann.

3. Bearbeiten Sie die Dämonkonfigurationsdatei so, dass sie mit Ihrer Datenquelle ausgeführt werden kann. Sie können die Konfigurationsdatei auch umbenennen.
  - Die erste Zeile in der Konfigurationsdatei muss aus einem Gleichheitszeichen bestehen. Fehlt das Gleichheitszeichen, wird der Dämon nicht gestartet. Eine Fehlermeldung meldet dann, dass der `DAEMON_PORT` nicht angegeben wurde.
  - Die letzte Zeile in der Konfigurationsdatei muss durch eine neue Zeile abgeschlossen werden. Die mit DB2 Information Integrator bereitgestellte Beispielkonfigurationsdatei endet mit einer neuen Zeile. Stellen Sie beim Bearbeiten der Datei sicher, dass die letzte Zeile in der Datei stets mit einer neuen Zeile abgeschlossen wird. Ist dies nicht der Fall, erhalten Sie eine Fehlermeldung bei dem Versuch, die erste BLAST-Abfrage unter Verwendung der in der letzten Zeile aufgeführten Datenquelle auszuführen.
  - Geben Sie die folgenden Optionen in der Konfigurationsdatei an. Bei Optionen, die einen Pfad erfordern, können relative Pfade angegeben werden. Relative Pfade beziehen sich auf das Verzeichnis, über das der Dämonprozess gestartet wurde.

#### **DAEMON\_PORT**

Hierbei handelt es sich um den Netzport, an dem der Dämon BLAST-Jobanforderungen empfängt, die vom Wrapper übergeben werden.

#### **MAX\_PENDING\_REQUESTS**

Hierbei handelt es sich um die maximale Anzahl an BLAST-Jobanforderungen, die für den Dämon zu einem bestimmten Zeitpunkt geblockt werden können. Es geht hier nicht um die Anzahl der BLAST-Jobs, die gleichzeitig ausgeführt werden, sondern lediglich um die Anzahl der Jobanforderungen, die zu einem bestimmten Zeitpunkt geblockt werden können. Es wird empfohlen, für diese Option einen Wert von über 5 anzugeben. Für den BLAST-Dämon gibt es keine Einschränkungen hinsichtlich der Anzahl der BLAST-Jobs, die gleichzeitig ausgeführt werden können.

#### **DAEMON\_LOGFILE\_DIR**

Hierbei handelt es sich um das Verzeichnis, in dem der Dämon die ihm zugeordnete Protokolldatei erstellt. Diese Datei enthält nützliche Status- und Fehlerinformationen, die vom BLAST-Dämon generiert werden.

#### **Q\_SEQ\_DIR\_PATH**

Hierbei handelt es sich um das Verzeichnis, in dem der Dämon eine temporäre Datendatei mit der Abfragesequenz erstellt. Der Inhalt dieser temporären Datei wird nach der Beendigung eines BLAST-Jobs gelöscht.

#### **BLAST\_OUT\_DIR\_PATH**

Hierbei handelt es sich um das Verzeichnis, in dem der Dämon die temporäre Datei zur Speicherung der BLAST-Ausgabedaten erstellt. Die Daten werden aus dieser Datei gelesen und über die Netzwerkverbindung zurück an den Wrapper übergeben. Nach der Übergabe der Daten an den Wrapper bereinigt der Dämon die temporäre Datei.

#### **BLASTALL\_PATH**

Hierbei handelt es sich um den vollständig qualifizierten Namen der ausführbaren BLAST-Datei auf der Maschine, die den Dämon ausführt.

#### **Datenbankspezifikationseintrag**

Dieser Eintrag gibt die Position einer BLAST-fähigen Datenquelle an. Notieren Sie sich die Datenbank *datenquellename*, die Sie in der Konfigurationsdatei angeben. Damit der Dämon ordnungsgemäß funktioniert, müssen Sie beim Erstellen des Kurznamens für die Datenquelle die Datenbank *datenquellename* angeben. Der Name ist abhängig von der Groß-/Kleinschreibung. Die Datenbank *datenquellename* wird in der Anweisung CREATE NICKNAME in der Option DATASOURCE angegeben.

Die Konfigurationsdatei muss mindestens einen Datenbankspezifikationseintrag in folgendem Format enthalten:

*datenquelle* = *pfad zur BLAST-fähigen\_datenquelle*

#### **Unter UNIX:**

Um beispielsweise die BLAST-fähige Datenquelle 'GenBank' anzugeben, fügen Sie der Konfigurationsdatei des Dämons die folgende Zeile hinzu:

```
genbank=/dsk/1/nuc1_data/genbank
```

### Unter Windows:

Um beispielsweise die BLAST-fähige Datenquelle 'GenBank' anzugeben, fügen Sie der Konfigurationsdatei des Dämons die folgende Zeile hinzu:

```
c:\vnr_data\genbank_nonest1.fasta
```

Der in einem Datenbankspezifikationseintrag angegebene Pfad muss die drei Indexdateien enthalten.

- Bei Datenquellen mit Informationen zu Nukleotiden haben die Indexdateien die folgenden Erweiterungen:
  - .nhr
  - .nin
  - .nsq
- Bei Datenquellen mit Informationen zu Aminosäuren haben die Indexdateien die folgenden Erweiterungen:
  - .phr
  - .pin
  - .psq

Der Datenbankspezifikationseintrag muss den Namen der Datei angeben, die die ursprünglichen Daten im FASTA-Format enthält. Die drei Indexdateien müssen denselben Rootnamen haben wie die Datei, die die ursprünglichen Daten im FASTA-Format enthält.

Die nächste Task in dieser Tasksequenz ist das Starten des BLAST-Dämons.

### Zugehörige Tasks:

- „Starten des BLAST-Dämons“ auf Seite 119

### Zugehörige Referenzen:

- „Konfigurationsdatei für den BLAST-Dämon - Beispiele“ auf Seite 118

## Konfigurationsdatei für den BLAST-Dämon - Beispiele

Folgende Beispiele zeigen den Inhalt einer Beispielkonfigurationsdatei.

### Beispiel – Konfigurationsdatei für den BLAST-Dämon (UNIX):

Dieses Beispiel zeigt die erforderlichen Optionen und Spezifikationen für GenBank und SWISS-PROT der BLAST-fähigen Datenquelle.

```
=
DAEMON_PORT=4007
MAX_PENDING_REQUESTS=10
DAEMON_LOGFILE_DIR=./
Q_SEQ_DIR_PATH=./
BLAST_OUT_DIR_PATH=./
BLASTALL_PATH=./blastall
genbank=/dsk/1/nucl_data/genbank
swissprot=/dsk/1/prot_data/swissprot
```

### Beispiel – Konfigurationsdatei für den BLAST-Dämon (Windows):

Dieses Beispiel zeigt die erforderlichen Optionen und Spezifikationen für GenBank und SWISS-PROT der BLAST-fähigen Datenquelle.

```

=
DAEMON_PORT=4007
MAX_PENDING_REQUESTS=10
DAEMON_LOGFILE_DIR=.
Q_SEQ_DIR_PATH=.
BLAST_OUT_DIR_PATH=.
BLASTALL_PATH=.blastall.exe
genbank=c:\vnr_data\genbank_nonest1.fasta
swissprot=c:\vnr_data\swissprot

```

#### Zugehörige Tasks:

- „Hinzufügen von BLAST-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 113
- „Konfigurieren des BLAST-Dämons“ auf Seite 115
- „Starten des BLAST-Dämons“ auf Seite 119

## Starten des BLAST-Dämons

Das Starten des BLAST-Dämons ist Teil der umfangreicheren Task des Hinzufügens von BLAST zu einem System zusammengeschlossener Datenbanken. Bevor Sie auf BLAST-Datenquellen zugreifen können, müssen Sie den BLAST-Dämon starten.

#### Voraussetzungen:

Bevor Sie den BLAST-Dämon starten, müssen Sie über Schreibzugriff auf alle Pfade verfügen, die unter den Einträgen DAEMON\_LOGFILE\_DIR, BLAST\_OUT\_DIR\_PATH und Q\_SEQ\_DIR\_PATH in der Konfigurationsdatei aufgeführt sind.

#### Vorgehensweise:

Gehen Sie wie folgt vor, um den BLAST-Dämon auf einer UNIX-Servermaschine zu starten:

1. Öffnen Sie das Verzeichnis, in dem die ausführbare Dämondatei sich befindet.
2. Geben Sie den Befehl `db2blast_daemon` aus.
  - Wenn Sie den Namen der Dämonkonfigurationsdatei nicht geändert haben und sich die Konfigurationsdatei in demselben Verzeichnis befindet wie die ausführbare Dämondatei, geben Sie den folgenden Befehl über die Befehlszeile ein:

```
db2blast_daemon
```

• Falls Sie den Namen der Dämonkonfigurationsdatei geändert haben oder sich die Dämonkonfigurationsdatei nicht in demselben Verzeichnis befindet wie die ausführbare Dämondatei, verwenden Sie im Dämonbefehl des Wrappers die Option `'-c'`, um die ausführbare Dämondatei auf den neuen Namen bzw. die neue Position zu verweisen.

Wird beispielsweise der folgende Befehl ausgegeben, sucht der Wrapperdämon die Konfigurationsdaten in einer Datei namens `BLAST_D.config` im Unterverzeichnis `cfg`.

```
db2blast_daemon -c cfg/BLAST_D.config
```

Die ausführbare Datei startet einen neuen Prozess, in dem der BLAST-Dämon ausgeführt wird.

Gehen Sie wie folgt vor, um den BLAST-Dämon auf einer Windows-Servermaschine zu starten:

1. Öffnen Sie das Verzeichnis, in dem die ausführbare Dämondatei sich befindet.

2. Geben Sie den Befehl `db2blast_daemon` mit den erforderlichen Parametern aus. Um beispielsweise den Dämons-service mit aktiviertem Debug zu installieren, geben Sie folgendes ein:

```
db2blast_daemon -a install -d 2
db2blast_daemon -a start
```

Die nächste Task in dieser Tasksequenz ist das Registrieren des BLAST-Wrappers.

#### Zugehörige Tasks:

- „Registrieren des BLAST-Wrappers“ auf Seite 121

## Befehl `db2blast_daemon` - Syntax und Beispiele

| Der Befehl `db2blast_daemon` kann auf UNIX- und Windows-Servern verwendet  
| werden. Einige der in der Syntax aufgelisteten Argumente können nur auf Win-  
| dows-Servern verwendet werden.

Die Syntax für den Befehl `db2blast_daemon` lautet wie folgt:

```
db2blast_daemon -a aktion -c konfigurationsdatei -d debugstufe
                  -u benutzer-id -p kennwort
```

#### -a *aktion*

Führt die angegebene Aktion aus. Gültige Aktionen sind *status*, *install*, *start*, *stop* und *remove*.

| Sie können dieses Argument nur auf Windows-Servern angeben.

#### -c *konfigurationsdatei*

Weist den Dämons-service zur Verwendung der angegebenen Konfigurationsdatei an. Geben Sie die Konfigurationsdatei nicht an, sucht der Dämon die Datei `BLAST_DAEMON.config` in dem Verzeichnis, in dem die ausführbaren Dämondateien installiert sind. Sie können diese Option mit den Aktionen *install* und *start* verwenden.

| Dieses Argument kann für UNIX- und Windows-Server angegeben werden.

#### -d *debugstufe*

Setzt die Debugstufe des Dämons-service auf den angegebenen Wert. Gültige Werte sind 1, 2 oder 3. Sie können diese Option mit den Aktionen *install* und *start* verwenden.

| Dieses Argument kann für UNIX- und Windows-Server angegeben werden.

#### -u *benutzer-id*

Legt für den Dämons-service die Ausführung unter der angegebenen Benutzer-ID fest. Sie können diese Option mit der Aktion *install* verwenden.

| Sie können dieses Argument nur auf Windows-Servern angeben.

#### -p *kennwort*

Gibt das Kennwort für die angegebene Benutzer-ID an. Das Kennwort ist nur dann gültig und erforderlich, wenn Sie die Option '-u' angeben. Ist beim Festlegen der Option '-u' die Option '-p' nicht angegeben, fordert Sie das Programm zur Eingabe des Kennworts auf. Sie können diese Option mit der Aktion *install* verwenden.

| Sie können dieses Argument nur auf Windows-Servern angeben.

Die mit der Aktion *start* angegebenen Optionen wirken sich nur auf die aktuelle Ausführung des Dämons aus und überschreiben die Werte, die mit der Aktion *install* angegeben werden.

### Beispiele:

Die folgenden Beispiele zeigen Dämonaktionen. In diesen Beispielen wird davon ausgegangen, dass die Datei `BLAST_DAEMON.config` sich in demselben Verzeichnis befindet wie die `db2blast_daemon.exe`.

- Geben Sie Folgendes ein, um den Status des Dämons zu prüfen:  
`db2blast_daemon -a status`
- Geben Sie Folgendes ein, um den Dämonservice mit aktiviertem Debug zu installieren:  
`db2blast_daemon -a install -d 2`
- Geben Sie Folgendes ein, um den Dämon zu starten:  
`db2blast_daemon -a start`
- Geben Sie Folgendes ein, um den Dämon zu stoppen:  
`db2blast_daemon -a stop`
- Geben Sie Folgendes ein, um den Dämonservice zu entfernen:  
`db2blast_daemon -a remove`

### Zugehörige Tasks:

- „Starten des BLAST-Dämons“ auf Seite 119

## Registrieren des BLAST-Wrappers

Die Registrierung des BLAST-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von BLAST-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf BLAST-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung `CREATE WRAPPER` mit dem Namen des Wrappers und dem Namen der Bibliotheksdatei des Wrappers aus.

Um beispielsweise einen Wrapper mit dem Namen `blast_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER blast_wrapper LIBRARY 'libdb2lsblast.a';
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung `CREATE WRAPPER` finden Sie in der Liste der Bibliotheksdateien von BLAST-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinition für den BLAST-Wrapper.

### Zugehörige Referenzen:

- „Bibliotheksdateien des BLAST-Wrappers“ auf Seite 122
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien des BLAST-Wrappers

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den BLAST-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, werden 3 Bibliotheksdateien zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden folgende Wrapperbibliotheksdateien zum Verzeichnispfad hinzugefügt: `libdb2lsblast.a`, `libdb2lsblastF.a` und `libdb2lsblastU.a`.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 24. Positionen und Dateinamen von BLAST-Wrapperbibliotheken*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	<code>/usr/opt/db2_08_01/lib/</code>	<code>libdb2lsblast.a</code>
Linux	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2lsblast.so</code>
Solaris	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2lsblast.so</code>
Windows	<code>%DB2PATH%\bin</code>	<code>db2lsblast.dll</code>

`%DB2PATH%` ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist `C:\Program Files\IBM\SQLLIB`.

### Zugehörige Tasks:

- „Registrieren des BLAST-Wrappers“ auf Seite 121

## Registrieren des Servers für BLAST-Datenquellen

Die Registrierung des Servers für eine BLAST-Datenquelle ist Teil der umfangreicheren Task des Hinzufügens von BLAST zu einem System zusammengeschlossener Datenbanken. Nach der Registrierung des Wrappers muss ein entsprechender Server registriert werden.

### Vorgehensweise:

Um den BLAST-Server für das System zusammengeschlossener Datenbanken zu registrieren, verwenden Sie die Anweisung `CREATE SERVER`.

Auf jeder Maschine in Ihrer Umgebung, auf der das ausführbare Programm BLAST und der Dämon installiert sind, müssen Sie einen Server für jeden BLAST-Suchtyp registrieren, den Sie unter Verwendung des betreffenden Exemplars des ausführbaren BLAST-Programms und Dämons ausführen wollen.

Wenn Sie beispielsweise den Server `blast_server1` für den mit der Anweisung `CREATE WRAPPER` erstellten Wrapper `my_blast` registrieren möchten, um ihn für BLASTn-Suchen zu verwenden, übergeben Sie die folgende Anweisung:



```
CREATE SERVER blast_server1
  TYPE blastn
  VERSION 2.1.2
  WRAPPER my_blast
  OPTIONS (NODE 'big_rs.company.com', DAEMON_PORT '4007')
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für BLAST-Datenquellen.

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für BLAST-Datenquellen“ auf Seite 123

**Zugehörige Referenzen:**

- „ALTER SERVER statement“ in *SQL Reference, Volume 2*
- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- „CREATE SERVER-Anweisungsargumente - BLAST-Wrapper“ auf Seite 602

## Registrieren von Kurznamen für BLAST-Datenquellen

### Registrieren von Kurznamen für BLAST-Datenquellen

Das Registrieren der Kurznamen für BLAST-Datenquellen ist Teil der umfangreicheren Task des Hinzufügens von BLAST zu einem System zusammengesetzter Datenbanken.

Nach der Registrierung eines Servers muss ein entsprechender Kurzname registriert werden. Kurznamen werden in Abfragen zum Verweisen auf BLAST-Datenquellen verwendet.

**Vorgehensweise:**

Um einen BLAST-Kurznamen zu registrieren, verwenden Sie die Anweisung `CREATE NICKNAME`

Da jeder BLAST-Suchtyp von einem eigenen Server verarbeitet wird, müssen Sie für jeden BLAST-Suchtyp, den Sie für eine BLAST-fähige Datenquelle ausführen wollen, einen eigenen Kurznamen definieren.

Beim Erstellen des Kurznamens geben Sie Spalteninformationen für den Definitionszeilenabschnitt der Datenquelle an. Alle anderen Spalten sind fixiert.

Dies war die letzte Task in dieser Tasksequenz.

**Zugehörige Konzepte:**

- „Syntaxanalyse von Definitionszeilen“ auf Seite 124

**Zugehörige Tasks:**

- „Hinzufügen von BLAST-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 113
- „Angabe von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75

**Zugehörige Referenzen:**

- „CREATE NICKNAME-Anweisungssyntax - BLAST-Wrapper“ auf Seite 603



- „Anweisung CREATE NICKNAME - Beispiele für den BLAST-Wrapper“ auf Seite 130
- „Fixierte Spalten für BLAST-Kurznamen“ auf Seite 124

## Syntaxanalyse von Definitionszeilen

Die Definitionszeile ist wie ein Schlüssel für jede Sequenz in der BLAST-fähigen Datenquelle und wird als Teil jeder mit BLAST gefundenen Übereinstimmung (Treffer) zurückgegeben. Die Definitionszeile wird auch als *define* bezeichnet.

Der Wert, der für eine Definitionszeile vom BLAST-Wrapper zurückgegeben und syntaktisch analysiert wird, ist nicht immer mit der Definitionszeile in der ursprünglichen FASTA-Datei identisch. Wenn z. B. Daten im Feld für die Zugriffsnummer eines BLAST-Treffers vorhanden sind, enthält die zurückgegebene Definitionszeile die Daten der Zugriffsnummer, gefolgt von den Daten des Definitionsfelds. Der Wrapper führt dann eine Syntaxanalyse für die Daten aus, die zurückgegeben werden.

**Empfehlung:** Um zu ermitteln, wie der Wrapper die Definitionszeile zurückgibt und syntaktisch analysiert, erstellen Sie einen Kurznamen mit einer einzigen Definitionszeilenspalte. Führen Sie anschließend eine Abfrage durch, um das Format anzuzeigen, das vom Wrapper der Definitionszeile für Ihre Datenquelle zurückgegeben wird.

Um die Informationen der Definitionszeile in Ihre Ergebniszeile einzuschließen, müssen Sie die Definitionsspalten in der Anweisung CREATE NICKNAME angeben. Jede angegebene Spalte muss die Optionen INDEX und DELIMITER enthalten. Wenn die letzte Spalte den Rest der Definitionszeileninformationen enthalten soll, können Sie die Option DELIMITER für die letzte angegebene Spalte auslassen.

Gültige Datentypspalten für die Definitionsspalten sind CLOB, DOUBLE, FLOAT, INTEGER und VARCHAR.

### Zugehörige Konzepte:

- „Benutzerdefinierte Funktion für die Syntaxanalyse von Definitionszeilen - Übersicht“ auf Seite 514

### Zugehörige Tasks:

- „Registrieren von Kurznamen für BLAST-Datenquellen“ auf Seite 123

### Zugehörige Referenzen:

- „CREATE NICKNAME-Anweisungssyntax - BLAST-Wrapper“ auf Seite 603
- „Anweisung CREATE NICKNAME - Beispiele für den BLAST-Wrapper“ auf Seite 130
- „Fixierte Spalten für BLAST-Kurznamen“ auf Seite 124

## Fixierte Spalten für BLAST-Kurznamen

Wenn Sie die Anweisung CREATE NICKNAME für eine BLAST-Datenquelle ausgeben, wird automatisch ein Satz fixierter Spalten mit dem Kurznamen erstellt.

Die fixierten Spalten sind ein Teil der Definition für den Kurznamen und werden im Katalog des Systems zusammengeschlossener Datenbanken erstellt. Sie können in SQL-Abfragen auf die fixierten Spalten verweisen. Es gibt zwei Typen von fixierten Spalten: fixierte Eingabespalten und fixierte Ausgabespalten.

**Fixierte Eingabespalten für BLAST-Kurznamen:** Fixierte Eingabespalten werden in der Klausel WHERE angegeben. Eingabespalten werden in SQL-Abfragen als Vergleichselemente verwendet, die Parameter übergeben. Sie übergeben BLAST-Standardswitches an BLAST. Anschließend wird BLAST für die angegebene Datenquelle ausgeführt, wobei diese Switches verwendet werden. Auf fixierte Eingabespalten kann auch in der Abfrageauswahlliste (SELECT-Liste) verwiesen werden. Diese Spalten werden als Teil der Ergebnistabelle zurückgegeben.

Die folgende Tabelle listet die fixierten Spalten auf, die in der Klausel WHERE verwendet werden können.

*Tabelle 25. Fixierte Eingabespalten für BLAST-Kurznamen*

Name	Datentyp	Operatoren	Beschreibung
BlastSeq	VARCHAR (32000) oder CLOB	=	Übergibt die Abfragesequenz an den BLAST-Wrapper.
E_Value	DOUBLE	<	Sowohl Eingabe- als auch Ausgabeparameter. Als Eingabeparameter gibt diese Spalte für den BLAST-Wrapper die Obergrenze der erwarteten Werte an, die von 'blastall' zurückgegeben werden sollten.
QueryStrands	INTEGER	=	Gibt an, welche Stränge bei der Ausführung einer BLASTn-Suche verglichen werden sollen. Der Wert 1 gibt an, dass der oberste Strang zu verwenden ist, 2 gibt den untersten Strang an, und 3 gibt an, dass beide Stränge verglichen werden sollen.
GapAlign	CHAR(1)	=	Gibt für den Wrapper an, ob in der BLAST-Ausgabe Alignments mit Lücken zulässig sind, d. h. ein DNA-Vergleich und die Suche nach DNA-Ähnlichkeiten mit Lücken (Gaps).
Matrix	VARCHAR(50)	=	Legt fest, welche Substitutionsmatrix von 'blastall' verwendet wird, um den Grad der Ähnlichkeit zwischen Aminosäurepaaren zu ermitteln. Dieses Vergleichselement wird nur von solchen BLAST-Suchtypen verwendet, die Aminosäuren mit Aminosäuren vergleichen.

Tabelle 25. Fixierte Eingabespalten für BLAST-Kurznamen (Forts.)

Name	Datentyp	Operatoren	Beschreibung
NMismatchPenalty	INTEGER	=	Gibt den Wert an, den 'blastall' vom Score eines Alignments abzieht, wenn eines der Nukleotidpaare im homologen Bereich nicht übereinstimmt. Dieses Vergleichselement wird nur von solchen BLAST-Suchtypen verwendet, die Nukleotide mit Nukleotiden vergleichen.
NMatchReward	INTEGER	=	Gibt den Wert an, den 'blastall' zum Score eines Alignments für jedes der Nukleotidpaare addiert, das im homologen Bereich übereinstimmt. Dieses Vergleichselement wird nur von solchen BLAST-Suchtypen verwendet, die Nukleotide mit Nukleotiden vergleichen.
FilterSequence	CHAR(1)	=	Gibt für 'blastall' an, ob eine Filterung durchgeführt werden soll, um biologisch irrelevante Segmente aus der Abfragesequenz zu entfernen. Beim Suchtyp BLASTn wird der Filter DUST verwendet. Ansonsten wird die Filterung von SEG ausgeführt.
NumberOfAlignments	INTEGER	=	Gibt an, wie viele HSP-Alignments in die BLAST-Ausgabe eingeschlossen werden sollen.
GapCost	INTEGER	=	Gibt den Wert an, den 'blastall' vom Score eines Alignments abzieht, wenn eine Lücke (Gap) entweder in die Abfragesequenz oder die Sequenz der Übereinstimmungen eingefügt werden muss, um eine Verlängerung des Alignments zu ermöglichen.
ExtendedGapCost	INTEGER	=	Gibt den Wert an, den 'blastall' vom Score eines Alignments abzieht, wenn eine Lücke (Gap), die entweder bereits in die Abfragesequenz oder die Sequenz der Übereinstimmungen eingefügt worden ist, um ein Nukleotid bzw. eine Aminosäure erweitert werden muss, um eine Verlängerung des Alignments zu ermöglichen.
WordSize	INTEGER	=	Gibt für 'blastall' die Länge der anfänglichen Übereinstimmungen an, die 'blastall' anfangs in der Datenbank sucht.

Tabelle 25. Fixierte Eingabespalten für BLAST-Kurznamen (Forts.)

Name	Datentyp	Operatoren	Beschreibung
ThresholdEx	INTEGER	=	Gibt den Schwellenwert des Scores an, bei dessen Unterschreitung BLAST nicht weiter versucht, die Übereinstimmung zu erweitern.

Sie können den Standarddatentyp für eine Spalte überschreiben, wenn Sie einen Kurznamen erstellen. Einige Spalten können große Datenmengen zurückgeben, z. B. die Spalten 'HSP\_H\_Seq' und 'HSP\_Midline'. Zur Rückgabe der ersten 50 Byte der Spalte können Sie die Spalte mit dem Datentyp VARCHAR(50) definieren. Nur die ersten 50 Byte werden dann in die Ausgabespalte kopiert.

**BLAST-Suchtypen und -Switches für fixierte Eingabespalten:** Die unterstützten BLAST-Suchtypen und -Switches für die einzelnen fixierten Eingabespalten werden in der folgenden Tabelle aufgeführt.

Tabelle 26. Von den fixierten Eingabespalten unterstützte BLAST-Suchtypen und -Switches

Name	BLAST-Suchtypen	BLAST-Switch	Erforderlich	Standardwert
BlastSeq	n, p, x, tn, tx	-l	Ja	n/v
E_Value	n, p, x, tn, tx	-e	Nein	10
QueryStrands	n	S	Nein	3
GapAlign	n, p, x, tn, tx	-g	Nein	T
Matrix	p, x, tn, tx	-n	Nein	BLOSUM62
NMisMatchPenalty	n	-q	Nein	-3
NMatchReward	n	-r	Nein	1
FilterSequence	n, p, x, tn, tx	-F	Nein	T
NumberOfAlignments	n, p, x, tn, tx	-b	Nein	250
GapCost	n, p, x, tn, tx	-G	Nein	11
ExtendedGapCost	n, p, x, tn, tx	-E	Nein	1
WordSize (für Blastn sind Werte unter 7 ungültig)	n, p, x, tn, tx	-W	Nein	11 -BLASTn 3 -BLASTp
ThresholdEx	n, p, x, tn, tx	-f	Nein	0

**Fixierte Ausgabespalten für BLAST-Kurznamen:** Die folgende Tabelle listet die fixierten Spalten auf, die in der Klausel WHERE verwendet werden können.

Tabelle 27. Fixierte Ausgabespalten für BLAST-Kurznamen

Name	Datentyp	Beschreibung
Score	DOUBLE	Der berechnete Score für die in den BLAST-Ergebnissen gemeldeten HSPs.
E_value	DOUBLE	Sowohl Eingabe- als auch Ausgabe-parameter. Als Ausgabe-parameter enthält diese Spalte den berechneten Score für die in den BLAST-Ergebnissen gemeldeten HSPs.

Tabelle 27. Fixierte Ausgabespalten für BLAST-Kurznamen (Forts.)

Name	Datentyp	Beschreibung
Length	INTEGER	Die Länge der in den BLAST-Ergebnissen gemeldeten Sequenz der Übereinstimmungen.
HIT_NUM	INTEGER	Die in den BLAST-Ergebnissen gemeldete Trefferzahl, beginnend mit 1.
HSP_NUM	INTEGER	Die Anzahl der in den BLAST-Ergebnissen gemeldeten HSPs, beginnend mit 1.
HSP_Info	VARCHAR(100)	Die Informationszeichenfolge für die von BLAST jeweils gemeldeten HSPs. Diese Zeichenfolge enthält Informationen über die Anzahl der Nukleotide bzw. Aminosäuren, die jeweils in der Abfragesequenz und der Sequenz der Übereinstimmungen identisch sind.
HSP_ALIGNMENT_LENGTH	INTEGER	Die Länge des HSP-Alignments.
HSP_IDENTITY	INTEGER	Die prozentuale Identität des Alignments, definiert als Anzahl der Identitäten dividiert durch die Länge des Alignments.
HSP_GAPS	INTEGER	Der Prozentsatz der Abstände in dem Alignment, definiert als Anzahl der Abstände dividiert durch die Länge des Alignments.
HSP_POSITIVE	INTEGER	Der Prozentsatz der positiven Zahlen des Alignments, definiert als die Anzahl der positiven Zahlen dividiert durch die Länge des Alignments.
HSP_QUERY_FRAME	INTEGER	Der Leserahmen des Alignments in der Abfragesequenz.  Nur verfügbar für die BLAST-Servertypen BLASTx, tBLASTn und tBLASTx.
HSP_HIT_FRAME	INTEGER	Der Leserahmen des Alignments in der Treffersequenz.  Nur verfügbar für die BLAST-Servertypen BLASTx, tBLASTn und tBLASTx.
HSP_Q_Start	INTEGER	Die numerische Position des ersten identischen Nukleotids bzw. der ersten identischen Aminosäure in der Abfragesequenz.
HSP_Q_End	INTEGER	Die numerische Position des letzten identischen Nukleotids bzw. der letzten identischen Aminosäure in der Abfragesequenz.

Tabelle 27. Fixierte Ausgabespalten für BLAST-Kurznamen (Forts.)

Name	Datentyp	Beschreibung
HSP_Q_Seq	VARCHAR(32000)	Das Segment der Abfragesequenz, das bei HSP_Q_Start beginnt und bei HSP_Q_End endet.  Sie können den Standarddatentyp für diese Spalte überschreiben und dafür CLOB mit einer maximalen Länge von 5 Megabyte angeben.
HSP_H_Start	INTEGER	Die numerische Position des ersten identischen Nukleotids bzw. der ersten identischen Aminosäure in der Sequenz der Übereinstimmungen.
HSP_H_End	INTEGER	Die numerische Position des letzten identischen Nukleotids bzw. der letzten identischen Aminosäure in der Sequenz der Übereinstimmungen.
HSP_H_Seq	VARCHAR(32000)	Das Segment der Sequenz der Übereinstimmungen, das bei HSP_H_Start beginnt und bei HSP_H_End endet.  Sie können den Standarddatentyp für diese Spalte überschreiben und dafür CLOB mit einer maximalen Länge von 5 Megabyte angeben.
HSP_Midline	VARCHAR(32000)	Die von BLAST ausgegebene Zeichenfolge, die den Grad der Übereinstimmung zwischen den Aminosäuren bzw. Nukleotiden an den einzelnen Positionen in den homologen Bereichen der Abfragesequenz und der Sequenz der Übereinstimmungen angibt.  Sie können den Standarddatentyp für diese Spalte überschreiben und dafür CLOB mit einer maximalen Länge von 5 Megabyte angeben.

**Zugehörige Konzepte:**

- „Syntaxanalyse von Definitionszeilen“ auf Seite 124

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für BLAST-Datenquellen“ auf Seite 123

**Zugehörige Referenzen:**

- „CREATE NICKNAME-Anweisungssyntax - BLAST-Wrapper“ auf Seite 603
- „Anweisung CREATE NICKNAME - Beispiele für den BLAST-Wrapper“ auf Seite 130

## Anweisung CREATE NICKNAME - Beispiele für den BLAST-Wrapper

Mit der folgenden Anweisung CREATE NICKNAME wird der Kurzname 'genbank' definiert.

Es wird davon ausgegangen, dass das Definitionsfeld in einem BLAST-Ergebnis folgende Informationen enthält:

```
>276342 15:8924 PMON5426
```

Hierbei gilt:

**276342** Das Zugriffsfeld des BLAST-Ergebnisses.

**15:8924 PMON5426**

Das Definitionsfeld in einem BLAST-Ergebnis, das die Nummer eines Organismus enthält, gefolgt von der Nummer eines Experiments und einer eindeutigen ID.

Anhand dieser Informationen wird der folgende Kurzname erstellt:

```
CREATE NICKNAME genbank (  
  acc_num integer  OPTIONS(INDEX '1', DELIMITER ' '),  
  org_num integer  OPTIONS(INDEX '2', DELIMITER ':'),  
  exp_num integer  OPTIONS(INDEX '3', DELIMITER ' '),  
  u_id  varchar(10) OPTIONS(INDEX '4')  
  FOR SERVER blast_server1  
  OPTIONS(DATASOURCE 'genbank', TIMEOUT '300');
```

Die Spalte acc\_num würde den Wert 276342 enthalten, die Spalte org\_num den Wert 15, die Spalte exp\_num den Wert 8924 und die Spalte u\_id den Wert PMON5426.

Nach der Übergabe der Anweisung CREATE NICKNAME können Sie den Kurznamen 'genbank' für Abfragen für Ihr System zusammenschlossener Datenbanken verwenden. Darüber hinaus können Sie den Kurznamen 'genbank' mit anderen Kurznamen und Tabellen im System zusammenschlossener Datenbanken verknüpfen.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für BLAST-Datenquellen“ auf Seite 123

### Zugehörige Referenzen:

- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „CREATE NICKNAME-Anweisungssyntax - BLAST-Wrapper“ auf Seite 603

---

## Einrichten von TurboBlast zum Arbeiten mit dem BLAST-Wrapper

### Einschränkungen:

TurboBlast unterstützt bestimmte blastall-Befehlsoptionen nicht. Beispielsweise wird die Option für Alignments mit Lücken (Gaps) -g F nicht unterstützt. Geben Sie für den Wert der Spalte GapAlign in Ihrem BLAST-Kurznamen F an, generiert TurboBlast einen Fehler. Eine vollständige Liste der nicht unterstützten Optionen finden Sie im Handbuch *TurboBlast 2.0 User Guide*.

### Vorgehensweise:

Einrichten von TurboBlast zum Arbeiten mit dem BLAST-Wrapper:

1. Installieren und konfigurieren Sie den BLAST-Wrapper. Führen Sie zum Testen Ihrer Einrichtung eine Abfrage für eine BLAST-fähige Datenbank aus.
2. Der BLAST-Wrapper und TurboBlast unterstützen die Betriebssysteme AIX, Linux, Solaris und Windows. Der BLAST-Dämon ist jedoch auf Windows-Betriebssystemen nicht verfügbar. Sobald der BLAST-Dämon für Windows verfügbar ist, kann er mit TurboBlast unter diesen Betriebssystemen ausgeführt werden.
3. Installieren und konfigurieren Sie TurboBlast entsprechend den Anweisungen im Handbuch *TurboBlast 2.0 Installation and Reference Guide*. Sie können TurboBlast auf verschiedene Arten installieren und einrichten. Soll der BLAST-Wrapper mit TurboBlast ausgeführt werden, muss der TurboBlast-Client auf der Maschine mit dem BLAST-Dämon installiert und eingerichtet werden. Mit dem BLAST-Dämon kann der Befehl `tblastall` aufgerufen werden.
4. Testen Sie das TurboBlast-System nach der Installation und Konfiguration von TurboBlast. Folgen Sie den Anweisungen im Handbuch *TurboBlast 2.0 Installation and Reference Guide*.
5. Ändern Sie Ihre Datei `BLAST_DAEMON.config` wie folgt:
  - a. Geben Sie den Parameter `BLASTALL_PATH` als den vollständigen Pfad von `tblastall` an. Beispiel: `BLASTALL_PATH=/home/blasttst/turboblast/Tblast-2.1/tblastall`
  - b. Geben Sie als Spezifikationseintrag der BLAST-fähigen Datenbank den Namen der BLAST-fähigen Datenbank an, den Sie zum Laden Ihrer BLAST-fähigen Datenbank in TurboBlast verwendet haben. Die Datenbanknamen werden angezeigt, wenn Sie in TurboBlast den Befehl `listdatabase -l` eingeben. Dieser TurboBlast-Datenbankname sollte anstatt des Pfads zu der BLAST-fähigen Datenquelle verwendet werden. Beispiel: `genbank=<der genbank-datenbankname in TurboBlast>`
6. Starten Sie den BLAST-Dämon erneut. Der BLAST-Dämon ruft den Befehl `tblastall` anstatt des Befehls `blastall` auf, um Suchvorgänge für BLAST-fähige Datenbanken auszuführen.
7. Die zu `tblastall` zugehörigen Protokolldateien werden in das Verzeichnis `DAEMON_LOGFILE_DIR` geschrieben, das in Ihrer Datei `BLAST_DAEMON.config` angegeben ist. Prüfen Sie auch die Protokolldateien `STDERR.log` und `STDOUT.log`, die vom BLAST-Dämon in demselben Verzeichnis generiert werden.

### Zugehörige Tasks:

- „Hinzufügen von BLAST-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 113
- „Konfigurieren des BLAST-Dämons“ auf Seite 115

---

## Aufbau von BLAST-SQL-Abfragen

Für die Übergabe von Standard-BLAST-Switches an die ausführbare Datei `'blastall'` werden Vergleichselemente für Eingabespalten verwendet. Vergleichselemente für die Ausgabespalten werden vom Server mit zusammengeschlossenen Datenbanken verarbeitet.

### Einschränkungen:

An den BLAST-Wrapper übergebene Abfragen sind nur dann gültig, wenn sie mindestens das Eingabevergleichselement `BlastSeq` enthalten. Alle anderen Vergleichselemente sind optional.



### Vorgehensweise:

Um eine BLAST-Abfrage aufzubauen, verwenden Sie die Eingabevergleichselemente in der WHERE-Klausel Ihrer SQL-Anweisung.

Das folgende Beispiel zeigt drei Eingabevergleichselemente: BlastSeq, GapCost und NMismatchPenalty.

```
Select * from blast b where  
BlastSeq = 'GTCCAGCC...' AND  
GapCost = -10 AND  
NMismatchPenalty = -4;
```

### Zugehörige Tasks:

- „Registrieren von Kurznamen für BLAST-Datenquellen“ auf Seite 123

### Zugehörige Referenzen:

- „BLAST-Datenquelle - Beispielabfragen“ auf Seite 132

---

## BLAST-Datenquelle - Beispielabfragen

Nachstehend werden einige BLAST-Beispielabfragen aufgeführt, um zu zeigen, wie Abfragen für BLAST-Datenquellen aufgebaut werden.

Verwenden Sie die Beispiele als Anleitung für die Ausführung von Abfragen.

Bei diesen Abfragen gibt der Name der einzelnen Kurznamen jeweils den BLAST-Suchtyp und die Datenquelle an. Dadurch brauchen die Registrierungsanweisungen nicht für jede Beispielabfrage aufgeführt zu werden. Außerdem verwenden einige der Abfragen andere hypothetische Datenquellen, sodass diese Beispiele das Verhalten des Wrappers beim Verknüpfen mit anderen Datenquellen darstellen können.

### Abfrage 1

```
select *  
from blastn_genbank  
where BlastSeq =  
'caaccctccagccgagttgtcaatggcgaggaagctgttccccac';
```

Wird diese SQL-Anweisung ausgeführt, führt der Wrapper anhand der angegebenen Sequenz eine BLASTn-Suche für GenBank aus. Der Wrapper gibt alle verfügbaren Spalten zurück, einschließlich der Eingabeparameterspalten sowie der BLAST-Ergebnisspalten.

### Abfrage 2

```
select *  
from blastn_genbank  
where BlastSeq =  
'caaccctccagccgagttgtcaatggcgaggaagctgttccccac'  
and GapCost = 8 and NMismatchPenalty = -4;
```

Wird diese SQL-Anweisung ausgeführt, führt der Wrapper anhand der angegebenen Sequenz eine BLASTn-Suche für GenBank aus. Außerdem übergibt der Wrapper die beiden angegebenen Parameter an den Dämon und die Befehlszeile 'blast-all'. Der Wrapper gibt alle verfügbaren Spalten zurück, einschließlich der Eingabeparameterspalten sowie der BLAST-Ergebnisspalten.

### Abfrage 3

```
select blp.*
from blastp_swissprot blp, protein_db prdb
where prdb.keyword = 'malic enzyme'
and blp.BlastSeq = prdb.sequence;
```

Wird diese SQL-Abfrage ausgeführt, führt der Wrapper keine oder mehrere BLASTp-Suchen für SWISS-PROT durch, je nach Anzahl der Sequenzen, die von einer hypothetischen Datenbank für Proteinsequenzen zurückgegeben werden. Diese Anweisung wird von DB2 in zwei getrennte Abfragen aufgeteilt, und für jede von der hypothetischen Proteindatenbank zurückgegebene Zeile wird eine BLASTp-Suche ausgeführt. Der Wrapper gibt alle verfügbaren Spalten zurück, einschließlich der Eingabeparameterspalten sowie der BLAST-Ergebnisspalten.

### Abfrage 4

```
select Score, E_Value, HSP_Info, HSP_Q_Seq, HSP_H_Seq, HSP_Midline
from blastx_swissprot
where BlastSeq = 'gagttgtcaatggcgagg'
and GapCost = 8;
```

Wird diese SQL-Anweisung ausgeführt, führt der Wrapper anhand der angegebenen Sequenz eine BLASTx-Suche für SWISS-PROT aus. In diesem Fall setzt 'blastall' die Eingabesequenz in allen sechs Leserahmen um und führt anhand der sechs neu erstellten Proteinsequenzen jeweils eine Suche nach Übereinstimmungen durch. Die HSPs in den Ergebnissen enthalten eher Aminosäure-Aminosäure-Alignments anstatt Nukleotid-Nukleotid-Alignments. Der angegebene Parameter wird an den Dämon und anschließend an 'blastall' über die Befehlszeile übergeben. Der Wrapper gibt nur diejenigen Spalten zurück, die in der Abfrage speziell angefordert wurden.

### Abfrage 5

```
select tblx.Score, tblx.E_Value, tblx.HSP_Info tblx.HSP_Q_Seq,
HSP_H_Seq, HSP_Midline
from tblastx_genbank tblx, gen_exp_database gedb
where tblx.BlastSeq = gedb.sequence
and gedb.organism = 'interesting organism'
and GapCost = 8
and FilterSequence = 'F';
```

Wird diese SQL-Abfrage ausgeführt, führt der Wrapper keine oder mehrere tBLASTx-Suchen für GenBank durch, je nach Anzahl der Sequenzen, die von einer hypothetischen Datenbank für Genausdrücke zurückgegeben werden. Diese Anweisung wird von DB2 in zwei getrennte Abfragen aufgeteilt, und für jede von der hypothetischen Datenbank der Genausdrücke zurückgegebene Zeile wird eine tBLASTx-Suche ausgeführt. In diesem Fall setzt 'blastall' die Eingabesequenz und alle GenBank-Sequenzen in allen sechs Leserahmen um und führt anhand der sechs neu erstellten Proteinabfragesequenzen und aller neu erstellten Proteinsequenzen in der Datenbank jeweils eine Suche nach Übereinstimmungen durch. Die HSPs in den Ergebnissen enthalten eher Aminosäure-Aminosäure-Alignments anstatt Nukleotid-Nukleotid-Alignments. Die angegebenen Parameter werden an den Dämon und anschließend an 'blastall' über die Befehlszeile übergeben. Der Wrapper gibt nur diejenigen Spalten zurück, die in der Abfrage speziell angefordert wurden.

### Zugehörige Referenzen:

- „Documentum-Datenquelle - Beispielabfragen“ auf Seite 212
- „Excel-Datenquelle - Beispielabfragen“ auf Seite 254

---

## Tipps für die Optimierung des BLAST-Wrappers

Zur Verbesserung der Netzwerkübertragungsleistung sollten sich der Server mit zusammengeschlossenen Datenbanken und der BLAST-Server auf verschiedenen Hardwareeinheiten befinden. Der BLAST-Dämon sollte sich auf dem BLAST-Server befinden.

### Zugehörige Tasks:

- „Konfigurieren des BLAST-Dämons“ auf Seite 115

---

## Nachrichten für den BLAST-Wrapper

In diesem Abschnitt sind die Nachrichten und ihre zugehörigen Beschreibungen aufgeführt, die bei der Verwendung des BLAST-Wrappers angezeigt werden können.

*Tabelle 28. Vom BLAST-Wrapper ausgegebene Nachrichten*

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "sqlno_crule_save_plans [100]:rc (-2144272209) Leere Planliste festgestellt".)	Die an DB2 übergebene SQL-Abfrage konnte vom Wrapper nicht verarbeitet werden. Korrigieren Sie die Syntax und übergeben Sie die Abfrage erneut.
SQL1816N	Der Wrapper "BLAST_WRAPPER" kann nicht für den Zugriff auf den "Typ" der Datenquelle (" <code>&lt;servertyp&gt;</code> "") verwendet werden, den Sie für die zusammengeschlossene Datenbank zu definieren versuchen.	In der Anweisung CREATE SERVER wurde ein ungültiger Typ (TYPE) verwendet. Der Typ muss einem der unterstützten BLAST-Typen entsprechen.
SQL1817N	Die Anweisung CREATE SERVER gibt die "Version" der Datenquelle nicht an, die Sie für die zusammengeschlossene Datenbank definieren möchten.	In der Anweisung CREATE SERVER wurde nicht die Version angegeben.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "Blast-Wrapper" empfangen. Zugeordneter Text und Token sind "Verbindung zum Dämon kann nicht hergestellt werden".	Der BLAST-Wrapper konnte keine Verbindung zum Dämon herstellen. Möglicherweise ist der Dämon nicht aktiv oder er wurde falsch konfiguriert oder die Maschine, auf der der Dämon ausgeführt wird, ist nicht erreichbar.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "Blast-Wrapper" empfangen. Zugeordneter Text und Token sind "Zeitlimit für Blast-Dämon ist abgelaufen".	Vor Ablauf des in der Anweisung CREATE NICKNAME angegebenen Zeitlimits wurden vom Dämon keine Ergebnisse empfangen. Erhöhen Sie das Zeitlimit oder überprüfen Sie den Dämon auf mögliche Probleme.

Tabelle 28. Vom BLAST-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "Blast-Wrapper" empfangen. Zugeordneter Text und Token sind "Blast-Dämon ist fehlgeschlagen".	Der Dämon hat die Kommunikation gestoppt, oder die zurückgegebenen Ergebnisse waren nicht ordnungsgemäß formatiert.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "Blast-Wrapper" empfangen. Zugeordneter Text und Token sind "Unbekannter Fehler vom Blast-Dämon".	Der Blast-Wrapper empfing vom Dämon einen Fehlercode, der nicht verarbeitet werden kann. Möglicherweise ist die Version des Dämons mit der Version des Wrappers nicht kompatibel.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "Blast-Wrapper" empfangen. Zugeordneter Text und Token sind "Umbenennung der Spalte ist nicht zulässig".	Es wurde eine Anweisung ALTER NICKNAME ausgegeben, um eine der Spalten umzubenennen. Das Umbenennen von Spalten ist jedoch nicht zulässig.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "Blast-Wrapper" empfangen. Zugeordneter Text und Token sind "XML-Parser-Fehler".	Der Xerces-Parser befindet sich in einem ungültigen Status oder hat eine Ausnahmere bedingung ausgegeben.
SQL1823N	Für den Datentyp "<datentypname>" gibt es vom Server "<servername>" keine Datentypzuordnung.	Der angegebene Datentyp wird von dieser Spalte nicht unterstützt.
SQL1881N	"DEFAULT" ist keine gültige Option "COLUMN" für "<spaltenname>".	Die Option DEFAULT wurde für eine Spalte verwendet, die diese Option nicht unterstützt. Spalten, die nur für Ausgabe verwendet werden, und Definitionszeilenspalten verfügen nicht über Standardwerte.
SQL1882N	Die Option "DEFAULT" der Art "COLUMN" kann nicht auf "<optionswert>" für "<spaltenname>" eingestellt werden.	Der für die Option DEFAULT angegebene Wert gehört zu einem Typ, der mit der Spalte nicht kompatibel ist, oder wurde nicht ordnungsgemäß formatiert.

#### Zugehörige Konzepte:

- „Nachrichten - Einführung“ in *Fehlernachrichten Band 1*

#### Zugehörige Referenzen:

- „SQLSTATE-Nachrichten“ in *Fehlernachrichten Band 2*



---

## Kapitel 8. Konfigurieren des Zugriffs auf Geschäftsanwendungsdatenquellen

In diesem Abschnitt wird erklärt, wie Geschäftsanwendungsdatenquellen Ihrem System zusammengeschlossener Datenbanken mit Hilfe des WebSphere Business Integration-Wrappers hinzugefügt werden.

---

### Der WebSphere Business Integration-Wrapper

Der WebSphere® Business Integration-Wrapper ist eine Wrapper mit Lesezugriff, der die WebSphere Business Integration-Adapter verwendet, um auf Geschäftsanwendungen zuzugreifen. Das Handbuch *IBM DB2 Information Integrator Systeme zusammengeschlossener Datenbanken* enthält eine Liste der unterstützten Adapter und Anwendungen.

Der WebSphere Business Integration-Wrapper stellt eine SQL-Schnittstelle zu Geschäftsanwendungen (wie beispielsweise Geschäftsanwendungen von SAP, Siebel und PeopleSoft) zur Verfügung. Mit dem WebSphere Business Integration-Wrapper können Sie die Funktionen von Systemen zusammengeschlossener Datenbanken verwenden, um Geschäftsdaten aus Geschäftsanwendungen mit Daten in anderen zusammengeschlossenen Datenquellen zu verknüpfen. Der WebSphere Business Integration-Wrapper extrahiert Geschäftsobjektdefinitionen in eine Hierarchie von Kurznamen.

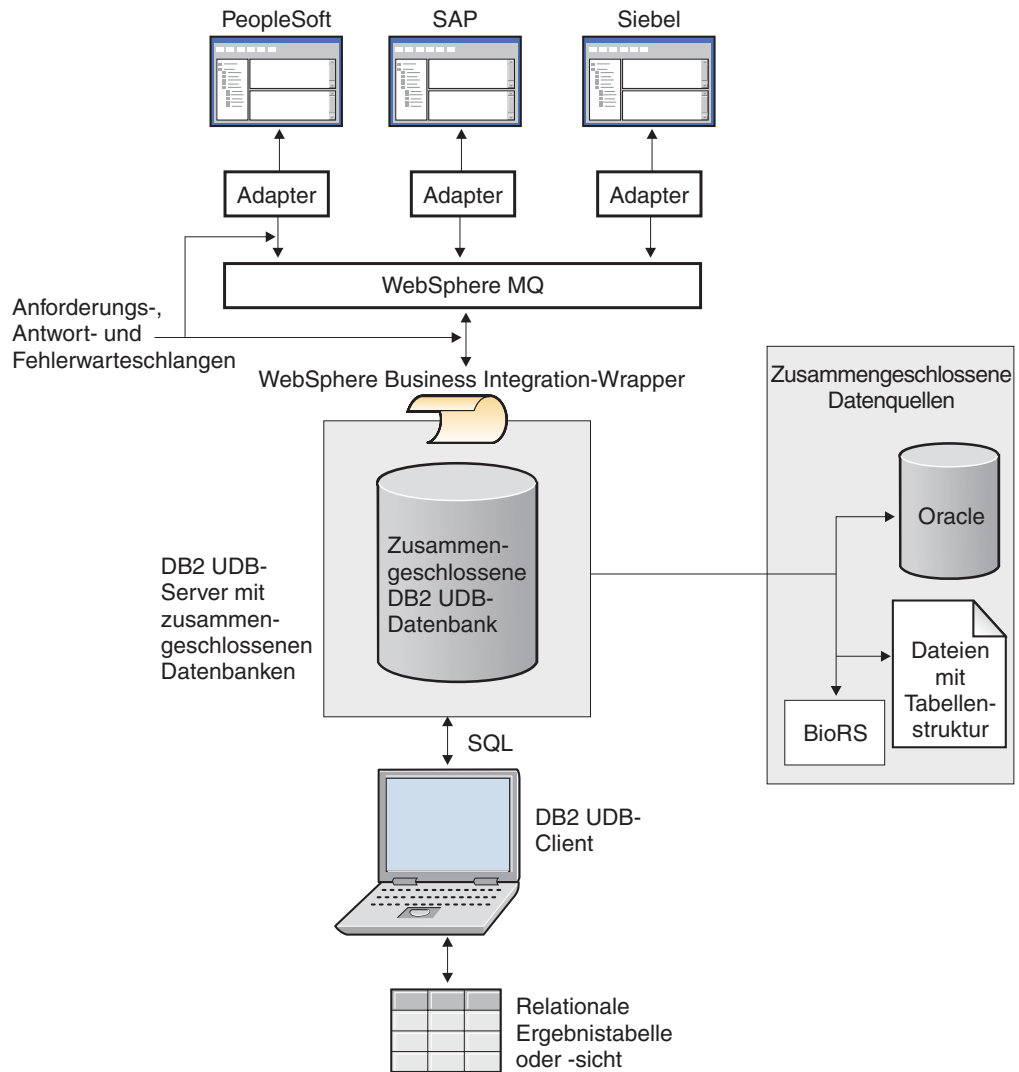


Abbildung 9. Der WebSphere Business Integration-Wrapper in der DB2® Universal Database-Umgebung

Abb. 9 zeigt die Beziehung zwischen dem WebSphere Business Integration-Wrapper und den Adaptern in der DB2 Universal Database™-Umgebung. Anhand der folgenden Schritte wird beschrieben, wie in einem System zusammengeschlossener Datenbanken auf Geschäftsanwendungsdaten zugegriffen wird:

1. Ein Benutzer sendet eine Abfrage an den Server mit zusammengeschlossenen Datenbanken, um auf einen Kurznamen zuzugreifen, der einer Datenquelle wie beispielsweise einer Siebel-Anwendung zugeordnet ist.
2. Der Wrapper setzt die Abfrage in ein Geschäftsobjekt um.
3. Der Wrapper stellt das Geschäftsobjekt in eine WebSphere MQ-Nachrichtenswarteschlange.
4. Der WebSphere Business Integration-Adapter für die betreffende Anwendung liest das Geschäftsobjekt (d. h. die Anforderung) aus der Nachrichtenswarteschlange.
5. Der WebSphere Business Integration-Adapter bereitet zusammen mit der Geschäftsanwendung ein Antwortgeschäftsobjekt vor.
6. Der WebSphere Business Integration-Adapter stellt das Antwortgeschäftsobjekt in die Nachrichtenswarteschlange.

7. Der Wrapper liest das Antwortgeschäftsobjekt aus der Antwortwarteschlange.
8. Der Wrapper extrahiert das Antwortgeschäftsobjekt in eine Ergebnismenge, die auf dem relationalen Schema basiert, das mit der Kurznamendefinition definiert wird.

**Zugehörige Konzepte:**

- „Geschäftsobjektdefinitionen“ auf Seite 139

**Zugehörige Tasks:**

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengesetzter Datenbanken“ auf Seite 144
- „Registrieren des WebSphere Business Integration-Wrappers“ auf Seite 145
- „Konfigurieren des WebSphere Business Integration-Adapters“ auf Seite 141
- „Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen“ auf Seite 148

**Zugehörige Referenzen:**

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175
- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158

---

## Geschäftsobjektdefinitionen

Eine Geschäftsobjektdefinition ist eine Schablone, über die der WebSphere® Business Integration-Adapter ein Exemplar eines Geschäftsobjekts erstellt. Eine Geschäftsobjektdefinition stellt eine Datenentität einer Geschäftsanwendung dar.

Ein Geschäftsobjekt ist ein Exemplar einer Geschäftsobjektdefinition und kann entweder als unstrukturierte oder hierarchische Struktur vorliegen. Ein unstrukturiertes Geschäftsobjekt enthält lediglich einfache Attribute. Ein hierarchisches Geschäftsobjekt enthält mindestens ein komplexes Attribut. Ein Repository von Geschäftsobjektdefinitionen ist in jedem WebSphere Business Integration-Adapter für die jeweils unterstützte Anwendung vorhanden.

Das folgende Beispiel zeigt ein unstrukturiertes Geschäftsobjekt:

**Kunde**

Gomez  
Juanita  
Apt 2C  
123 Main Street  
Big City  
California  
91234  
888  
1111111

Das folgende Beispiel zeigt ein hierarchisches Geschäftsobjekt mit einigen komplexen Attributen:

**Ansprechpartner** (übergeordnetes Element)

ID  
Kunden-ID  
Datum  
Text  
Autorisierung  
Positionen (es gibt 0 oder mehr Positionselemente)



(untergeordnete Elemente)  
Geschäftsobjekt 1  
Geschäftsobjekt 2  
Geschäftsobjekt 3

Geschäftsobjektdefinitionen müssen mit Hilfe des ODA-Tools (Object Discovery Agent = Objekterkennungsagent) erstellt werden. Hierbei handelt es sich um ein Tool, das zusammen mit jedem WebSphere Business Integration-Adapter geliefert wird. Das ODA-Tool generiert eine XML-Schemadefinitionsdatei für eine Geschäftsobjektdefinition. Unter Umständen generiert dieser Agent mehrere Schemadateien, wenn das definierte Geschäftsobjekt eine hierarchische Struktur aufweist.

Die XML-Schemadefinition ist eine Datei vom Dateityp `.xsd`. Sie befindet sich in einem Verzeichnis, das in der WebSphere Business Integration-Konfiguration angegeben wird. Die Geschäftsobjektdefinition muss vor dem Erstellen der Kurznamen für den WebSphere Business Integration-Wrapper generiert werden. Weitere Informationen zur Familie der WebSphere Business Integration-Adapter finden Sie auf folgender Website: [www.ibm.com/websphere/integration/wbiadapters](http://www.ibm.com/websphere/integration/wbiadapters).

Zum Erstellen von Kurznamen verwenden Sie die `.xsd`-Datei, die vom ODA-Tool erstellt wird. Kurznamen liefern eine relationale Schemadarstellung der Geschäftsobjektdefinition. Der WebSphere Business Integration-Wrapper ordnet ein hierarchisches Geschäftsobjekt einer Hierarchie relationaler Kurznamen zu. Beispiel: Jedes untergeordnete Geschäftsobjekt mit der Kardinalität 'n' wird einem eigenen Kurznamen zugeordnet, der mit dem Kurznamen des übergeordneten Geschäftsobjekts mit einer Fremdschlüsselintegritätsbedingung verknüpft ist.

Die WebSphere Business Integration-Geschäftsobjekte, auf die IBM® DB2® Information Integrator zugreifen kann, werden den spezifischen Anwendungsentitäten in folgender Tabelle zugeordnet:

*Tabelle 29. Geschäftsobjekte und die zugehörigen Anwendungsentitäten*

Geschäftsobjekte	Anwendungsentitäten
Siebel	Geschäftskomponente (Business Component)
PeopleSoft	Komponentenschnittstelle (Component Interface)
SAP	BAPI

**Zugehörige Konzepte:**

- „Der WebSphere Business Integration-Wrapper“ auf Seite 137

**Zugehörige Tasks:**

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengesetzter Datenbanken“ auf Seite 144
- „Konfigurieren des WebSphere Business Integration-Adapters“ auf Seite 141

**Zugehörige Referenzen:**

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175

## Konfigurieren des WebSphere Business Integration-Adapters

Für jede Geschäftsanwendung, auf die Sie über SQL-Anweisungen mit Hilfe der Wrapperfunktionen für zusammengeschlossene Datenbanken zugreifen wollen, müssen Sie jeweils einen WebSphere Business Integration-Adapter installieren und konfigurieren. Jeder Adapter wird einer Definition eines Servers mit zusammengeschlossenen Datenbanken zugeordnet.

### Voraussetzungen:

- Lesen Sie im Handbuch *IBM DB2 Information Integrator Systeme zusammengeschlossener Datenbanken* die Liste der unterstützten Adapter.
- Lesen Sie unter IBM WebSphere Business Integration Information Center die Informationen zur Installation der jeweiligen Adapter.
- Lesen Sie die Dokumentation zu den WebSphere Business Integration-Adaptoren mit den Hilfetexten zu den jeweiligen Geschäftsanwendungen.
- Installieren Sie sämtliche aktuellen Fixpacks für den jeweiligen Adapter, den Sie verwenden wollen. Die relevanten Unterstützungsinformationen für die Adapter erhalten Sie auf der Website WebSphere Business Integration Adapters Support.
- Lesen Sie das Handbuch *WebSphere MQ System Administration Guide* mit den Informationen zum Konfigurieren der Nachrichtenwarteschlangen.
- Lesen Sie die Installationsinformation unter IBM WebSphere Business Integration Information Center mit den Informationen zu den Adaptern und den Konfigurationsmerkmalen.

### Vorgehensweise:

Um einen WebSphere Business Integration-Adapter zu konfigurieren, gehen Sie wie folgt vor:

1. Konfigurieren Sie das ODA-Tool (Object Discovery Agent = Objekterkennungsagent) und das BOD-Tool (Business Object Designer = Geschäftsobjekt designer) und erstellen Sie die Geschäftsobjektdefinitionen.  
Wenn Sie die Geschäftsobjektdefinitionen im BOD-Tool konfigurieren, geben Sie je nach Geschäftsanwendung die folgenden Werte für Verben an:

Tabelle 30. Werte für Verben nach Geschäftsanwendungen

Geschäftsanwendung	Verb
SAP	Retrieve
PeopleSoft	Retrieve
Siebel	RetrieveByContent

Weitere Informationen zur Konfiguration und Verwendung eines ODA-Tools enthält die Dokumentation des Adapters, den Sie konfigurieren.

2. Verwenden Sie das CC-Tool (Connector Configurator = Konnektorkonfigurationsprogramm) über die Schnittstelle des WebSphere Business Integration-Adapters, um eine Konfigurationsdatei zu definieren, die folgende Informationen enthält:
  - Die vom Adapter unterstützten Geschäftsobjekte.
  - Die Konfigurationsmerkmale für den Adapter. Es gibt sowohl standardmäßige als auch anwendungsspezifische Konfigurationsmerkmale.

### Standardmäßige Konfigurationsmerkmale

Sie müssen einige Merkmalwerte anpassen, um den Adapter mit IBM DB2 Information Integrator verwenden zu können. Die nachstehende Liste enthält einige der spezifischen Merkmale, die konfiguriert werden müssen:

- Geben Sie den Wert des Integrationsbrokers als WMQI an.
- Geben Sie die Speicherposition des Metadatenrepositorys an, dessen Eigner der Adapter ist. Die XML-Schemadefinitionsdateien mit den Geschäftsobjektdefinitionen werden an dieser Position gespeichert.
- Geben Sie den Typ der Datenzustellungstransport als WMQI-MQ an.
- Geben Sie den Namen des Warteschlangenmanagers an, der die vom Adapter verwendeten Warteschlangen verwaltet.
- Geben Sie die Namen der acht Warteschlangen an, die für die Ausführung des Adapters erforderlich sind.

### Anwendungsspezifische Konfigurationsmerkmale

Mit diesen Merkmalen werden Werte für bestimmte, anwendungsspezifische Komponenten angegeben. Mit Hilfe der von Ihnen zur Verfügung gestellten Werte wird eine Sitzung mit der betreffenden Anwendung aufgebaut. Darüber hinaus steuern diese Merkmale das Verarbeitungsverhalten für die anwendungsspezifischen Komponenten.

3. Definieren Sie die drei WebSphere MQ-Nachrichtenwarteschlangen, die für den Wrapper erforderlich sind: 'request\_queue' (Anforderungswarteschlange), 'response\_queue' (Antwortwarteschlange) und 'fault\_queue' (Fehlerwarteschlange).

WebSphere MQ ist die Nachrichtenübertragungs- und Transportschicht zwischen dem Adapter und den Wrappern.

#### **request\_queue (Anforderungswarteschlange)**

Stellt die Anforderungsnachrichten von DB2 Information Integrator an den Adapter zu.

#### **response\_queue (Antwortwarteschlange)**

Stellt die Antwortnachrichten vom Adapter an DB2 Information Integrator zu.

#### **fault\_queue (Fehlerwarteschlange)**

Stellt die Fehlernachrichten vom Adapter an DB2 Information Integrator zu. Der Adapter stellt eine Nachricht in diese Warteschlange, wenn er nicht in der Lage ist, die Nachricht in die Warteschlange für zu beantwortende Nachrichten zu stellen.

Bei diesen Warteschlangen handelt es sich um statische Warteschlangen, die verwendet werden, um Nachrichten (einschließlich Datenobjekten und Fehler-  
nachrichten) zwischen dem Adapter und dem Wrapper auszutauschen.

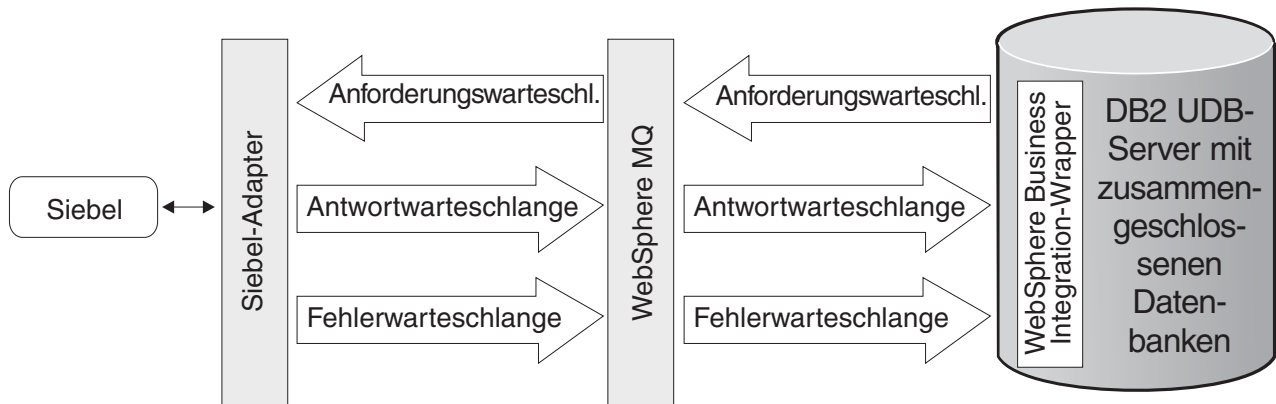


Abbildung 10. Topologie der WebSphere-Nachrichtenwarteschlangen für den Transport von Informationen zwischen den Siebel-Geschäftsanwendungen und dem DB2-Server mit zusammengeschlossenen Datenbanken

4. Definieren Sie die fünf zusätzlichen Nachrichtenwarteschlangen, die für den Adapter erforderlich sind:

- AdminInQueue
- AdminOutQueue
- SynchronousRequestQueue
- SynchronousResponseQueue
- DeliveryQueue

Die WebSphere Business Integration-Adapter benötigen fünf zusätzliche Warteschlangen, die verwendet werden, wenn der jeweilige Adapter nicht mit DB2 Information Integrator, sondern mit einem WMQI-Broker eingesetzt wird. Sie müssen diese zusätzlichen Nachrichtenwarteschlangen erstellen und konfigurieren, um den Adapter starten zu können.

5. Definieren Sie die WebSphere MQ-Benutzerberechtigung anhand einer der folgenden Methoden:

- Definieren Sie die ID des DB2-Exemplareigners als Teil der MQManager-Gruppe.
- Stellen Sie sicher, dass der MQManager-Administrator beim Erstellen des Kanals 'ServerConnection' einen Wert für MCAUSER setzt. Der Wert für MCAUSER muss eine Benutzer-ID sein, die zur MQManager-Gruppe oder zur Gruppe der Administratoren gehört.

**Zugehörige Konzepte:**

- „Geschäftsobjektdefinitionen“ auf Seite 139
- „Der WebSphere Business Integration-Wrapper“ auf Seite 137

**Zugehörige Tasks:**

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengeschlossener Datenbanken“ auf Seite 144
- „Registrieren des WebSphere Business Integration-Wrappers“ auf Seite 145

**Zugehörige Referenzen:**

- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158

---

## Hinzufügen von Geschäftsanwendungen zu einem Server mit zusammengeschlossenen Datenbanken

### Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengeschlossener Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Geschäftsanwendungsdatenquellen zu konfigurieren, müssen Sie dem Server Informationen zu den Datenquellen und -objekten zur Verfügung stellen, auf die Sie zugreifen möchten.

Der Zugriff des Servers mit zusammengeschlossenen Datenbanken auf Geschäftsanwendungsdatenquellen kann über die DB2-Steuerzentrale oder die DB2-Befehlszeile konfiguriert werden. Die DB2-Steuerzentrale umfasst einen Assistenten, der Sie durch die erforderlichen Schritte zum Konfigurieren des Servers mit zusammengeschlossenen Datenbanken führt.

#### Voraussetzungen:

- DB2 Information Integrator muss auf einem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank, die ein 32-Bit-Exemplar von DB2 UDB verwendet, muss auf dem Server mit zusammengeschlossenen Datenbanken vorhanden sein.

#### Vorgehensweise:

Um Geschäftsanwendungsdatenquellen einem System zusammengeschlossener Datenbanken hinzuzufügen, gehen Sie wie folgt vor:

1. Registrieren Sie den WebSphere Business Integration-Wrapper.
2. Registrieren Sie die Serverdefinition.
3. Registrieren Sie Kurznamen für Geschäftsanwendungsdatenquellen.
4. Optional: Erstellen Sie Sichten für zusammengeschlossene Datenbanken für die WebSphere Business Integration-Kurznamen.

#### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63
- „Geschäftsobjektdefinitionen“ auf Seite 139
- „Der WebSphere Business Integration-Wrapper“ auf Seite 137

#### Zugehörige Tasks:

- „Editieren der Oracle-Prozedur 'genclntsh' und Erstellen der Datei 'libclntsh' nach der Installation von DB2 Information Integrator (HP-UX, Linux, Solaris)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Registrieren des WebSphere Business Integration-Wrappers“ auf Seite 145
- „Registrieren der Serverdefinition für Geschäftsanwendungsdatenquellen“ auf Seite 146
- „Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen“ auf Seite 148

- „Erstellen von Sichten für zusammengeschlossene Datenbanken für Geschäftsanwendungskurznamen“ auf Seite 157
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43
- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Registrieren des WebSphere Business Integration-Wrappers“ auf Seite 145
- „Registrieren der Serverdefinition für Geschäftsanwendungsdatenquellen“ auf Seite 146
- „Erstellen von Sichten für zusammengeschlossene Datenbanken für Geschäftsanwendungskurznamen“ auf Seite 157
- „Konfigurieren des WebSphere Business Integration-Adapters“ auf Seite 141
- „Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen“ auf Seite 148

#### **Zugehörige Referenzen:**

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## **Registrieren des WebSphere Business Integration-Wrappers**

Beim Hinzufügen von Geschäftsanwendungsdatenquellen zu einem Server mit zusammengeschlossenen Datenbanken muss der WebSphere Business Integration-Wrapper registriert werden.

Die Registrierung eines Wrappers ist für den Zugriff auf Geschäftsanwendungsdatenquellen erforderlich. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken verwendet, um mit Datenquellen zu kommunizieren und Daten aus Datenquellen abzurufen. Wrapper werden als Gruppe von Bibliotheksdateien implementiert. Sie können den Wrapper über die DB2-Steuerzentrale oder die DB2-Befehlszeile registrieren. Die DB2-Steuerzentrale umfasst einen Assistenten, der Sie durch die erforderlichen Schritte zum Registrieren des Wrappers führt.

#### **Voraussetzungen:**

1. Installieren und konfigurieren Sie den entsprechenden Adapter.
2. Installieren und konfigurieren Sie WebSphere MQ Version 5.3 (CSD-Stufe 5).
3. Erstellen Sie die WebSphere MQ-Nachrichtenwarteschlangen.
4. Ist der WebSphere MQ-Manager nicht auf demselben System wie DB2 Information Integrator installiert, müssen Sie den Client von WebSphere MQ Version 5.3 (CSD-Stufe 5) auf demselben System installieren, auf dem ein Serverexemplar von DB2 Information Integrator installiert wurde.

#### **Vorgehensweise:**

Um einen Wrapper zu registrieren, setzen Sie die Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Wrapperbibliotheksdatei ab.

Um beispielsweise einen Wrapper mit dem Namen wbi\_wrapper auf dem Server mit zusammengeschlossenen Datenbanken zu registrieren, der das Betriebssystem Windows verwendet, setzen Sie die folgende Anweisung ab:

```
CREATE WRAPPER wbi_wrapper LIBRARY 'db2wbi.dll';
```

Der Name der von Ihnen angegebene Wrapperbibliotheksdatei hängt von dem Betriebssystem auf dem Server mit zusammengeschlossenen Datenbanken ab. Den

korrekten Namen, der in der Anweisung CREATE WRAPPER angegeben werden muss, finden Sie in der Liste der Bibliotheksdateien des WebSphere Business Integration-Wrappers.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den WebSphere Business Integration-Wrapper.

**Zugehörige Konzepte:**

- „Der WebSphere Business Integration-Wrapper“ auf Seite 137

**Zugehörige Tasks:**

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengesetzter Datenbanken“ auf Seite 144

**Zugehörige Referenzen:**

- „Bibliotheksdateien des WebSphere Business Integration-Wrappers“ auf Seite 146
- „Bibliotheksdateien des WebSphere Business Integration-Wrappers“ auf Seite 146

## Bibliotheksdateien des WebSphere Business Integration-Wrappers

Die folgende Tabelle enthält eine Liste der Verzeichnispfade und Bibliotheksdateinamen für den WebSphere Business Integration-Wrapper.

Wenn Sie DB2 Information Integrator installieren, werden dem in der Tabelle aufgeführten Verzeichnispfad 3 Bibliotheksdateien hinzugefügt. Wird der Server mit zusammengesetzten Datenbanken beispielsweise unter AIX ausgeführt, werden die Wrapperbibliotheksdateien libdb2wbi.a, libdb2wbiF.a und libdb2wbiU.a dem Verzeichnispfad hinzugefügt.

Wenn Sie einen Wrapper registrieren, geben Sie nur den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 31. Speicherpositionen und Dateinamen der WebSphere Business Integration-Wrapperbibliothek*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2wbi.a
Windows	%DB2PATH%\bin	db2wbi.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows lautet C:\Program Files\IBM\SQLLIB.

**Zugehörige Tasks:**

- „Registrieren des WebSphere Business Integration-Wrappers“ auf Seite 145

## Registrieren der Serverdefinition für Geschäftsanwendungsdatenquellen

Beim Hinzufügen einer Geschäftsobjektanwendung zu einem System zusammengesetzter Datenbanken muss die Serverdefinition für eine Geschäftsobjektdatei registriert werden.



Nach der Registrierung des Wrappers muss ein entsprechender Server registriert werden.

#### **Einschränkungen:**

Sie können die Option MQ\_SVRCONN\_CHANNELNAME nur dann angeben, wenn Sie die Option MQ\_CONN\_NAME angeben. Die Option MQ\_CONN\_NAME kann erst gelöscht werden, nachdem die Option MQ\_SVRCONN\_CHANNELNAME gelöscht wurde. Wird MQ\_CONN\_NAME nicht angegeben, verwendet das System zusammengeschlossener Datenbanken den Wert der Umgebungsvariablen MQSERVER. Die Umgebungsvariable MQSERVER wird in der Datei 'db2dj.ini' gesetzt. Wenn Sie die Datei 'db2dj.ini' editieren, müssen Sie DB2 Universal Database zunächst stoppen und anschließend erneut starten.

#### **Vorgehensweise:**

Um die Serverdefinition für eine Geschäftsanwendung im System zusammengeschlossener Datenbanken zu registrieren, setzen Sie die Anweisung CREATE SERVER ab.

Um beispielsweise eine Serverdefinition für die Siebel-Geschäftsanwendungen zu registrieren, müssen Sie folgende Anweisung absetzen:

```
CREATE SERVER siebel_server
  VERSION 2.4
  WRAPPER wbi_wrapper
  OPTIONS ( App_Type 'siebel',
           Request_Queue 'myqueue3',
           Response_Queue 'myqueue4',
           Fault_Queue 'myqueue5',
           MQ_Manager 'mymq',
           MQ_REPONSE_TIMEOUT '55000',
           MQ_CONN_NAME '9.30.76.151(1420)',
           MQ_SVRCONN_CHANNELNAME 'SYSTEM.DEF.SVRCONN'
  )
```

In vorstehendem Beispiel ist die Geschäftsanwendung eine Siebel-Anwendung, die mit der Option APP\_TYPE angegeben wird. Gültige Werte sind SIEBEL, PSOFT und SAP. Die Option VERSION gibt die Version des verwendeten WebSphere Business Integration-Adapters an. Gültige Werte sind 2.3 und 2.4. Die Serveroptionen müssen die Warteschlangendefinitionen umfassen, die im Abschnitt 'Konfigurieren des WebSphere Business Integration-Adapters' beschrieben werden. Die Standard-einstellung für MQ\_RESPONSE\_TIMEOUT lautet 50000 Millisekunden. Der Wert -1 gibt an, dass keine Zeitsperrenbegrenzung vorhanden ist.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für Geschäftsanwendungsdatenquellen.

#### **Zugehörige Konzepte:**

- „Der WebSphere Business Integration-Wrapper“ auf Seite 137

#### **Zugehörige Tasks:**

- „Konfigurieren des WebSphere Business Integration-Adapters“ auf Seite 141
- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengeschlossener Datenbanken“ auf Seite 144
- „Registrieren des WebSphere Business Integration-Wrappers“ auf Seite 145



- „Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen“ auf Seite 148

**Zugehörige Referenzen:**

- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641
- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158

## Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen

### Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen

Beim Hinzufügen von Geschäftsanwendungen zu einem System zusammenschlossener Datenbanken müssen Kurznamen für Geschäftsanwendungsdatenquellen registriert werden.

Sie können den Kurznamen über die DB2-Steuerzentrale oder die DB2-Befehlszeile registrieren. Die DB2-Steuerzentrale umfasst einen Assistenten, der Sie durch die erforderlichen Schritte zum Registrieren des Kurznamens führt.

Sie müssen Kurznamen erstellen, die der strukturellen Hierarchie Ihrer Geschäftsobjektdefinition entsprechen. Übergeordnete Kurznamen enthalten mindestens einen untergeordneten Kurznamen. Untergeordnete Kurznamen entsprechen den Elementen, die eine Kardinalität von größer als 1 enthalten und innerhalb des Elements für den übergeordneten Kurznamen verschachtelt sind.

**Vorgehensweise:**

Um Kurznamen für Geschäftsanwendungsdatenquellen über die DB2-Befehlszeile zu registrieren, setzen Sie die Anweisung CREATE NICKNAME ab.

Um beispielsweise den Kurznamen 'sieb\_ssa\_Contact\_Contact' für eine Siebel-Geschäftsobjektdefinition zu registrieren, setzen Sie die folgende Anweisung ab:

```
CREATE NICKNAME sieb_ssa_Contact_Contact_NN(
  Id VARCHAR(15)  OPTIONS(XPATH './ns1:Id/text()'),
                    TEMPLATE '<ns1:Id>&column</ns1:Id>'),
  FirstName VARCHAR(50) OPTIONS(XPATH './ns1:FirstName/text()'),
                    TEMPLATE '<ns1:FirstName>&column</ns1:FirstName>'),
  LastName VARCHAR(50)  OPTIONS(XPATH './ns1:LastName/text()'),
                    TEMPLATE '<ns1:LastName>&column</ns1:LastName>'),
  AccountId VARCHAR(255) OPTIONS(XPATH './ns1:AccountId/text()'),
  PrimaryAccountName VARCHAR(100)
                    OPTIONS(XPATH './ns1:PrimaryAccountName/text()'),
  PrimaryPostalCode VARCHAR(30)
                    OPTIONS(XPATH './ns1:PrimaryPostalCode/text()'),
  PrimaryStreetAddress VARCHAR(200)
                    OPTIONS(XPATH './ns1:PrimaryStreetAddress/text()'),
  SalesRep VARCHAR(255)  OPTIONS(XPATH './ns1:SalesRep/text()'),
  State VARCHAR(255)    OPTIONS(XPATH './ns1:State/text()')
  FOR SERVER siebel_server
  OPTIONS(XPATH '//ns1:sieb_ssa_Contact_Contact',
  TEMPLATE '<ns1:sieb_ssa_Contact_Contact>
            &Id[1,1] &FirstName[0,1] &LastName[0,1]
            </ns1:sieb_ssa_Contact_Contact>',
  BUSOBJ_NAME 'sieb_ssa_Contact_Contact',
```

```
NAMESPACES 'ns1="http://www.ibm.com/websphere/
crossworlds/2002/BOSchema/
sieb_ssa_Contact_Contact"');
```

Die Kurznamenoption BUSOBJ\_NAME gibt den Namen der XML-Schema-definitionsdatei (XSD-Datei) an, die die Geschäftsobjektdefinition darstellt.

### **Erforderlicher Schritt: Markierung der erforderlichen Eingabespalte in den XSD-Dateien.**

Die WebSphere Business Integration-Adapter können als Antwort auf eine Abrufanforderung lediglich ein einziges Geschäftsobjektexemplar zurückgeben. Falls infolge einer Abrufanforderung im Format eines Eingabegeschäftsobjekts an den Adapter mehr als ein Geschäftsobjekt in der Anwendung ermittelt wird, gibt der Adapter nur das erste Geschäftsobjekt zurück. Der Wrapper gibt eine Fehlermeldung aus, die darauf hinweist, dass nicht die vollständige Ergebnismenge abgerufen werden konnte.

Um sicherzustellen, dass als Antwort auf eine Abrufanforderung lediglich ein einziges Geschäftsobjekt ermittelt wird, müssen dem Adapter im Anforderungsgeschäftsobjekt ausreichend Vergleichselemente zur Verfügung gestellt werden. Der Wrapper muss alle Eingabevergleichselemente senden, die erforderlich sind, damit nur ein einziges Geschäftsobjekt ermittelt wird. Daher müssen die Spalten anhand der korrekten Schablonenverweise in den Kurznamendefinitionen identifiziert werden. In den nachstehenden Schritten werden die Aktionen beschrieben, die ausgeführt werden müssen, um die korrekten erforderlichen Eingabespalten zu identifizieren, bevor Kurznamendefinitionen über die DB2-Steuerzentrale generiert werden:

1. Ermitteln Sie diejenigen Spalten im SAP-, Siebel- oder PeopleSoft-Anwendungsrepository, die einen eindeutigen Schlüssel für die zuzuordnende Anwendungsentität darstellen.

**SAP** Sie können das SAP-Geschäftsobjektrepository (Business Object Repository) verwenden, um die erforderlichen Eingabeparameter für die BAPI zu ermitteln, die mit Hilfe des ODA-Tools (Object Discovery Agent = Objekterkennungsagent) von WebSphere Business Integration einer WebSphere Business Integration-Geschäftsobjektdefinition zugeordnet wird.

**Siebel** Verwenden Sie eine der folgenden Methoden:

- Die Siebel-Anwendung verfügt über eine eindeutige ID-Spalte, die der jeweiligen Geschäftskomponente (Business Component) zugeordnet ist, und generiert für diese Spalte Hexadezimalwerte für jedes Exemplar der Geschäftskomponente. Diese ID-Spalte befindet sich auf der höchsten Ebene der Geschäftskomponentenhierarchie und ist bereits durch die Angabe 'isKey="true"' (im Abschnitt 'appSpecific-Info' der Anmerkung 'xml') für das Element in der generierten .xsd-Datei markiert.
- Mit Hilfe von Siebel-Tools können Sie die Datenbankspalten ermitteln, die einen zusammengesetzten eindeutigen Schlüssel für die zuzuordnende Geschäftskomponente darstellen. Diese Spalten müssen sich alle auf der höchsten Ebene bzw. der Basisebene der Geschäftsobjekthierarchie befinden.

#### **PeopleSoft**

Mit Hilfe des AD-Tools (Application Designer = Anwendungsentwicklungsprogramm) können Sie die getKey-Spalten in der

Komponentenschnittstelle (Component Interface) für die höchste Ebene der Hierarchie ermitteln, die einer Geschäftsobjektdefinition von WebSphere Business Integration zugeordnet wird.

2. Editieren Sie die XSD-Dateien, die für die Geschäftsobjektdefinition vom ODA-Tool von WebSphere Business Integration generiert werden, um die erforderlichen Eingabespalten zu markieren. Die Richtlinien für das Markieren der Spalten befinden sich im Abschnitt 'Option TEMPLATE auf Kurznamen- und Spaltenebene'.
3. Generieren Sie die Kurznamen-DDL für die Geschäftsobjektdefinition über die DB2-Steuerzentrale.

Um Kurznamen für Geschäftsanwendungsdatenquellen über die DB2-Steuerzentrale zu registrieren, gehen Sie wie folgt vor:

1. Erweitern Sie den Ordner **Objekte zusammenschlossener Datenbanken**.
2. Erweitern Sie den Ordner des Wrappers, für den Sie Kurznamen registrieren wollen.
3. Erweitern Sie den Ordner **Serverdefinitionen**.
4. Erweitern Sie den Ordner des Servers, für den Sie Kurznamen registrieren wollen.
5. Klicken Sie den Ordner **Kurznamen** mit Maustaste 2 an und wählen Sie **Erstellen** aus.
6. Klicken Sie im Fenster 'Kurznamen erstellen' die Option **Aufspüren** an, um Suchkriterien für die Auswahl von Objekten in der Datenquelle zu definieren.
7. Geben Sie die XML-Schemadefinitionsdatei an, die die Definition der Geschäftsobjekte enthält, auf die Benutzer von DB2 Information Integrator Zugriff haben sollen.
8. Klicken Sie **OK** an, um den Kurznamen gemäß der ausgewählten XML-Schemadefinitionsdatei zu erstellen.

Die DB2-Steuerzentrale extrahiert die Schemadatei in mehrere Anweisungen zum Erstellen von Kurznamen-DDLs und verwendet hierbei die entsprechenden Definitionen für die Abhängigkeiten zwischen übergeordneten und untergeordneten Elementen. Die erstellten Kurznamen stellen die Geschäftsobjekthierarchie dar, die in der XML-Schemadefinitionsdatei definiert ist.

Optional: Die nächste Task in dieser Tasksequenz ist das Erstellen von Sichten für zusammenschlossene Datenbanken für die Geschäftsanwendungskurznamen.

#### **Zugehörige Konzepte:**

- „Option TEMPLATE auf Kurznamen- und Spaltenebene“ auf Seite 151
- „Geschäftsobjektdefinitionen“ auf Seite 139
- „Der WebSphere Business Integration-Wrapper“ auf Seite 137
- „Option TEMPLATE auf Kurznamen- und Spaltenebene“ auf Seite 151
- „Kurznamen und XPATH-Ausdrücke“ auf Seite 156

#### **Zugehörige Tasks:**

- „Angeben von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75
- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammenschlossener Datenbanken“ auf Seite 144
- „Registrieren des WebSphere Business Integration-Wrappers“ auf Seite 145

- „Erstellen von Sichten für zusammengeschlossene Datenbanken für Geschäftsanwendungskurznamen“ auf Seite 157

#### Zugehörige Referenzen:

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175
- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172

## Option TEMPLATE auf Kurznamen- und Spaltenebene

Dieser Abschnitt bezieht sich auf den WebSphere® Business Integration-Wrapper und den Web-Services-Wrapper.

Der WebSphere® Business Integration-Wrapper und der Web-Services-Wrapper erstellen XML-Dokumente, die für den WebSphere Business Integration-Adapter und die Web-Services-Umgebung erforderlich sind. Zum Zeitpunkt der Erstellung des Kurznamens benötigen die Wrapper die Schablonenfragmente auf Kurznamen- und Spaltenebene. Dies ist die Option TEMPLATE in der Anweisung CREATE NICKNAME. Die Wrapper verwenden diese Informationen während der Abfrageplanungsphase und der Abfrageausführungsphase.

### Web-Services-Wrapper

Beim Web-Services-Wrapper variieren die erforderlichen und optionalen Attribute gemäß den Definitionen im WSDL-Dokument und gemäß der Art und Weise, wie eine Spalte abgeleitet wird. Eine Spalte kann entweder von einem Element oder von einem Attribut eines Elements abgeleitet werden.

- Wird die Spalte von einem Element abgeleitet, wird mit dem Wert für das Attribut 'minOccurs' festgelegt, ob eine Spalte optional oder erforderlich ist.
- Ist für 'minOccurs' der Wert 0 angegeben, ist die Spalte optional.
- Ist für 'minOccurs' der Wert 1 angegeben, ist die Spalte erforderlich.
- Wird die Spalte von einem Attribut eines Elements abgeleitet, wird mit dem Wert für 'use' im Attribut festgelegt, ob die Spalte optional oder erforderlich ist.
- Enthält ein Attribut den Wert 'use=optional', ist die Spalte optional.
- Enthält ein Attribut den Wert 'use=required', ist die Spalte erforderlich.

Das folgende Beispiel zeigt ein Attribut in einer Schemadefinition, die einer Spalte zugeordnet ist:

```
<xsd:complexType>
  <xsd:sequence>
    <xsd:element ref="tns:ZooName"/>
    <xsd:element ref="tns:Count"/>
    <xsd:element ref="tns:LastModified"/>
    <xsd:element maxOccurs="unbounded" minOccurs="0" ref="tns:Zookeeper"/>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:string" use="optional"/>
</xsd:complexType>
```

### WebSphere Business Integration-Wrapper

Beim WebSphere Business Integration-Wrapper variieren die erforderlichen und optionalen Spalten gemäß der Anwendung und dem zugeordneten Adapter. Sie

müssen die erforderlichen und optionalen Eingabespalten identifizieren, indem Sie für diese Spalten die entsprechenden Werte für die Schablonenoption TEMPLATE angeben. Bevor Sie die Kurznamen mit Hilfe der DB2®-Steuerzentrale erstellen, müssen Sie die XML-Schemadefinitionsdatei modifizieren, um die erforderlichen und optionalen Eingabespalten entsprechend zu markieren.

### SAP-BAPI

Die IBM® DB2-Steuerzentrale legt die erforderlichen und optionalen Eingabespalten auf Grundlage des Wertes bestimmter Markierungen in den XML-Schemadefinitionsdateien (XSD-Dateien) fest, die die Geschäftsobjektdefinition darstellen.

Im Anmerknungsabschnitt ('annotation') eines Elements auf einer beliebigen Ebene der Geschäftsobjekthierarchie (übergeordnete oder untergeordnete Geschäftsobjekte) gibt das Präfix I im Wert für 'appSpecificInfo' einen Importparameter für die SAP-BAPI an, der die Geschäftsobjektdefinition zugeordnet ist. Das Präfix E gibt einen Exportparameter für die SAP-BAPI an. Einige Elemente können sowohl Import- als auch Exportparameter für eine BAPI sein. Das folgende Beispiel zeigt ein Element, bei dem es sich sowohl um einen Import- als auch einen Exportparameter handelt:

```
<bx:appSpecificInfo>ICOMPANYCODE:ECOMPANYCODE</bx:appSpecificInfo>
```

Die Präfixe werden vom ODA-Tool (Object Discovery Agent = Objekterkennungsagent) von WebSphere Business Integration automatisch generiert. Grundlage für die Generierung sind die Informationen, die aus dem SAP-Geschäftsobjektrepository extrahiert werden.

Wird ein Element, das einen Importparameter darstellt (Präfix I im Wert für 'appSpecificInfo'), mit dem Attribut minOccurs=1 angegeben, identifiziert die DB2-Steuerzentrale das Element als erforderlichen Eingabeparameter und markiert die Elemente in der Kurznamendefinition als erforderliche Eingabespalte. Das ODA-Tool von WebSphere Business Integration setzt den Wert von minOccurs für die erforderlichen Eingabeparameter der SAP-BAPI nicht automatisch auf 1. Sie müssen auf das SAP-Geschäftsobjektrepository verweisen, um alle erforderlichen Eingabeparameter für die BAPI zu ermitteln, auf die Sie zugreifen wollen. Anschließend müssen Sie die entsprechenden Elemente in der XML-Schemadatei editieren, indem Sie das Attribut minOccurs=1 manuell setzen. Wird für einen Eingabeparameter weiterhin die Standardeinstellung 0 für das Attribut minOccurs verwendet, gibt die DB2-Steuerzentrale die Spalte in der generierten Kurznamenhierarchie als optionale Eingabespalte an.

Das folgende Beispiel zeigt eine optionale Eingabespalte:

```
<xsd:element name="Company_code" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICOMPANYCODE:</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="true" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="4" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
```

Das folgende Beispiel zeigt eine erforderliche Eingabespalte:

```
<xsd:element name="Company_id" minOccurs="1">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICOMPANYID:</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="true" isKey="true" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="4" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
```

Die erforderlichen und optionalen Eingabespalten für SAP-Geschäftsanwendungen werden durch die in folgender Tabelle dargestellte Syntax ausgewiesen:

*Tabelle 32. Markierungsschema für SAP-Eingabespalteninformationen*

Verwendete Markierungen in SAP-XSD-Dateien	Erforderliche Eingabespalte	Spaltenverweis in Kurznamensschablone
Alle Elemente auf beliebiger Hierarchieebene mit Präfix = 'I' und 'minOccurs=1'	Ja	&spaltenname[1,1]
Alle Elemente auf beliebiger Hierarchieebene mit Präfix = 'O' und 'minOccurs=0'	Nein	&spaltenname[0,1]

### Siebel und PeopleSoft

Die DB2-Steuerzentrale legt die erforderlichen und optionalen Eingabespalten auf Grundlage des Vorhandenseins und des Wertes der Markierung `isRequired` im Abschnitt `attributeInfo` der Anmerkung `annotation` für das Element fest. Ist keine Markierung `isRequired` vorhanden, ist die Spalte keine erforderliche Eingabespalte. Das ODA-Tool von WebSphere Business Integration generiert diese Markierungen in der XSD-Datei nicht automatisch. Sie müssen die erforderlichen und optionalen Eingabespalten ermitteln und sie in der XSD-Datei entsprechend markieren, bevor Sie mit Hilfe der DB2-Steuerzentrale die Kurznamen-DDL generieren.

Das folgende Beispiel zeigt die Markierungen für eine erforderliche Eingabespalte und für optionale Eingabespalten in der XSD-Datei für eine Siebel- bzw. PeopleSoft-Geschäftsobjektdefinition.

---

```

<xsd:element name="sieb_ssa_Contact_Contact">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version="1.0.0">
        <bx:appSpecificInfo>ON=Contact;CN=Contact</bx:appSpecificInfo>
      </bx:boDefinition>
    </xsd:appinfo>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Id" minOccurs="0">
        <xsd:annotation>
          <xsd:appinfo>
            <bx:boAttribute>
              <bx:appSpecificInfo>FN=Id</bx:appSpecificInfo>
              <bx:attributeInfo isForeignKey="false"
                isKey="true" isRequired="true" />
            </bx:boAttribute>
          </xsd:appinfo>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

...

---

Abbildung 11. Abschnitt einer Siebel-Geschäftsobjektdefinition (Teil 1 von 2)

---

```

...
<xsd:element name="FirstName" minOccurs="1">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>FN=First Name</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false"
          isRequired="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="50" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
<xsd:element name="LastName" minOccurs="1">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>FN=Last Name</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false"
          isRequired="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...

```

---

Abbildung 11. Abschnitt einer Siebel-Geschäftsobjektdefinition (Teil 2 von 2)



Die erforderlichen und optionalen Eingabespalten für Siebel- und PeopleSoft-Geschäftsanwendungen werden durch die in folgender Tabelle dargestellte Syntax ausgewiesen:

*Tabelle 33. Markierungsschema für Siebel- und PeopleSoft-Eingabespalteninformationen*

Verwendete Markierungen in Siebel- und PeopleSoft-XSD-Dateien	Erforderliche Eingabespalte	Spaltenverweis in Kurznamen-schablone
isRequired="true"	Ja	&spaltenname[1,1]
isRequired="false"	Nein	&spaltenname[0,1]

Das folgende Beispiel zeigt die DDL, die von der DB2-Steuerzentrale auf Grundlage der XSD-Datei erstellt wird, die in der Abbildung mit der Bezeichnung *Abschnitt einer Siebel-Geschäftsobjektdefinition* dargestellt ist. Die XSD-Datei in dieser Abbildung enthält für das Attribut 'isRequired' den Wert 'false'.

```
CREATE NICKNAME sieb_ssa_contact_contact_nn(
  Id VARCHAR(15) OPTIONS(XPATH './ns1:Id/text()'),
    TEMPLATE '<ns1:Id>&column</ns1:Id>'),
  FirstName VARCHAR(50) OPTIONS(XPATH './ns1:FirstName/text()'),
    TEMPLATE '<ns1:FirstName>&column</ns1:FirstName>'),
  LastName VARCHAR(50) OPTIONS(XPATH './ns1:LastName/text()'),
    TEMPLATE '<ns1:LastName>&column</ns1:LastName>'),
  AccountId VARCHAR(255) OPTIONS(XPATH './ns1:AccountId/text()'),
  PrimaryAccountName VARCHAR(100)
    OPTIONS(XPATH './ns1:PrimaryAccountName/text()'),
  PrimaryPostalCode VARCHAR(30)
    OPTIONS(XPATH './ns1:PrimaryPostalCode/text()'),
  PrimaryStreetAddress VARCHAR(200)
    OPTIONS(XPATH './ns1:PrimaryStreetAddress/text()'),
  SalesRep VARCHAR(255) OPTIONS(XPATH './ns1:SalesRep/text()'),
  State VARCHAR(255) OPTIONS(XPATH './ns1:State/text()')
  FOR SERVER siebel_server
  OPTIONS(XPATH '//ns1:sieb_ssa_contact_contact',
    TEMPLATE '<ns1:sieb_ssa_contact_contact>
      &Id[1,1] &FirstName[0,1] &LastName[0,1]
    </ns1:sieb_ssa_contact_contact>',
  BUSOBJ_NAME 'sieb_ssa_contact_contact',
  NAMESPACE 'ns1="http://www.ibm.com/websphere/
    crossworlds/2002/BOSchema/sieb_ssa_contact_contact"');
```

#### Zugehörige Konzepte:

- „Der WebSphere Business Integration-Wrapper“ auf Seite 137
- „Der Web-Services-Wrapper und das WSDL-Dokument“ auf Seite 433

#### Zugehörige Tasks:

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengesetzter Datenbanken“ auf Seite 144
- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 439
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442



### Zugehörige Referenzen:

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175
- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158
- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Nachrichten für den Web-Services-Wrapper“ auf Seite 468
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172

### Kurznamen und XPATH-Ausdrücke

Dieser Abschnitt bezieht sich auf den WebSphere® Business Integration-Wrapper und den Web-Services-Wrapper.

Kurznamen entsprechen der Baumstruktur der Daten Ihres XML-Dokuments. Übergeordnete und untergeordnete Kurznamen entsprechen der Stammstruktur (Root) und den verschachtelten Elementen der Datenbaumstruktur. Diese übergeordneten und untergeordneten Kurznamen werden durch Primär- und Fremdschlüssel miteinander verbunden, die in der Anweisung CREATE NICKNAME angegeben werden.

Jeder Kurzname wird durch XPath-Ausdrücke definiert, die Ausgabewerte darstellen. Der WebSphere Business Integration-Wrapper und der Web-Services-Wrapper verwendet XPath-Ausdrücke, um eine Entsprechung zwischen den Daten in einem XML-Dokument und den Zeilen in einer relationalen Tabelle herzustellen. Diese XPath-Ausdrücke identifizieren die Werte im XML-Dokument und ermitteln, welche Entsprechung diese Werte zu den Spalten der einzelnen Zeilen haben. Die Daten im XML-Dokument werden vom WebSphere Business Integration-Wrapper und vom Web-Services-Wrapper lediglich gelesen. Die Wrapper führen keine Aktualisierung der Daten durch. Die Option XPATH enthält die Informationen zum Ermitteln der SOAP-Nachrichten über die Kennungen der SOAP-Rahmenanweisung und des SOAP-Hauptteils. Die Nachricht 'getQuote' ist in den Elementen der SOAP-Rahmenanweisung und des SOAP-Hauptteils enthalten.

Der XPATH-Ausdruck der Option NICKNAME zeigt auf sich wiederholende Kennungen, die sich im Ausgabeelement befinden. Der XPath-Ausdruck legt fest, wie viele bzw. welche Zeilen der Kurzname enthalten wird. Der XPATH-Ausdruck der Spaltenoption steht in einem relativen Verhältnis zum XPATH-Ausdruck NICKNAME. Der XPATH-Ausdruck der Spaltenoption identifiziert die Werte in einer Zeile. Ein XPATH-Ausdruck der Option NICKNAME in einem untergeordneten Kurznamen steht in einem relativen Verhältnis zu einem XPATH-Ausdruck der Option NICKNAME in einem übergeordneten Kurznamen.

Beim Erstellen eines Kurznamen werden Optionen ausgewählt, die die Zuordnung zwischen dem Kurznamen und dem XML-Dokument angeben. Kurznamen, die für WebSphere Business Integration-Wrapper erstellt werden, sind einem XML-Schemadefinitionsdokument (XSD-Dokument) zugeordnet. Kurznamen, die für Web-Services-Wrapper erstellt werden, sind einem WSDL-Dokument zugeordnet.

### **Zugehörige Konzepte:**

- „Was ist XML?“ auf Seite 471
- „Der Web-Services-Wrapper und das WSDL-Dokument“ auf Seite 433
- „Datenzuordnungen zwischen Kurznamen und XML-Dokumenten“ auf Seite 479

### **Zugehörige Tasks:**

- „Hinzufügen von XML zu einem System zusammenschlossener Datenbanken“ auf Seite 474
- „Registrieren von Kurznamen für XML-Datenquellen“ auf Seite 482
- „Erstellen von Sichten für zusammenschlossene Datenbanken für Nicht-Stammkurznamen (XML-Wrapper)“ auf Seite 487
- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammenschlossener Datenbanken“ auf Seite 144
- „Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen“ auf Seite 148
- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammenschlossenen Datenbanken“ auf Seite 439
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442

### **Zugehörige Referenzen:**

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175
- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158
- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172
- „Anweisung CREATE NICKNAME - Beispiele für den XML-Wrapper“ auf Seite 482

## **Erstellen von Sichten für zusammenschlossene Datenbanken für Geschäftsanwendungskurznamen**

Beim Hinzufügen von Geschäftsanwendungen zu einem System zusammenschlossener Datenbanken können Sichten für zusammenschlossene Datenbanken für Geschäftsanwendungskurznamen erstellt werden.

Sie haben die Möglichkeit, die Sichten für zusammenschlossene Datenbanken über die Hierarchie von Kurznamen definieren, die eine Geschäftsobjekthierarchie beschreiben. Durch das Definieren von Sichten für zusammenschlossene Datenbanken wird sichergestellt, dass die Abfragen, die Teile der Hierarchie der Geschäftsanwendungskurznamen verknüpfen, ordnungsgemäß ausgeführt werden.

### **Vorgehensweise:**

Um Sichten für zusammenschlossene Datenbanken für Geschäftsanwendungskurznamen zu erstellen, gehen Sie wie folgt vor:

1. Definieren Sie eine Sicht für jeden Geschäftsanwendungskurznamen als Verknüpfung aller Kurznamen in dem Pfad zum übergeordneten Kurznamen.
2. Definieren Sie in der Klausel WHERE der Sicht die Spalten PRIMARY\_KEY und FOREIGN\_KEY als Verknüpfungspredikate.

3. Integrieren Sie alle Spalten des Geschäftsanwendungskurznamens in die SELECT-Liste der Sicht, mit Ausnahme der Spalte, die mit der Kurznamenspaltenoption FOREIGN\_KEY ausgewiesen ist. Sie dürfen keine Spalten in die SELECT-Liste integrieren, die in den übergeordneten Kurznamen im Verlauf der Hierarchie als PRIMARY\_KEY ausgewiesen sind.
4. Integrieren Sie die erforderlichen Eingabespalten für die Hierarchie in die SELECT-Liste. Diese Spalten gehören unter Umständen zu einigen anderen Kurznamen in der Hierarchie.

Das folgende Beispiel zeigt eine Sicht, die auf Kurznamen basiert, die aus einem Geschäftsobjekt generiert wurden. Die Klausel WHERE enthält den Primär- und den Fremdschlüssel, die in einer Anweisung CREATE NICKNAME definiert sind.

```
CREATE VIEW view1 (
  customer, bankkey, bankact, customerno )
AS (SELECT b.customer, b.bank_key, b.bank_acct,
  a.customerno
  FROM sap_bapi_customer_getdetail2_NN a,
  sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
  WHERE a.NN_PKEY=b.NN_FKEY);
```

Abfragen, die die Sicht 'view1' verwenden, müssen Vergleichselementwerte für die erforderliche Spalte einschließen, wie aus folgendem Beispiel ersichtlich wird:

```
SELECT * FROM view1
  WHERE customerno='1234567890';
```

Dies war die letzte Task in dieser Tasksequenz.

#### Zugehörige Tasks:

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengesetzter Datenbanken“ auf Seite 144

#### Zugehörige Referenzen:

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175

## Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper

### Beispiel 1: Unstrukturiertes Geschäftsobjekt

Abb. 12 auf Seite 159 zeigt einen Abschnitt aus einer .xsd-Datei, die eine WebSphere Business Integration-Geschäftsobjektdefinition für eine Siebel-Geschäftskomponente darstellt. Die Hierarchie der Geschäftsobjektdefinition besteht aus einer einzigen Ebene, die lediglich das Stammgeschäftsobjekt enthält. Die DB2-Steuerzentrale erstellt zur Darstellung dieser Geschäftsobjektdefinition nur einen einzigen relationalen Kurznamen.

In der .xsd-Datei ist das ID-Element als erforderliche Eingabespalte markiert. Hierzu wurde die Markierung 'isRequired="true"' im Anmerkungsabschnitt (annotation) für das Element hinzugefügt. Die Spalten 'FirstName' und 'LastName' sind als optionale Eingabespalten ausgewiesen. Hierzu wurde die Markierung 'isRequired="false"' hinzugefügt.

---

```

<?xml version="1.0" encoding="utf-8" standalone="no"?>
<xsd:schema elementFormDefault="qualified"
  targetNamespace="http://www.ibm.com/websphere/
  crossworlds/2002/BOSchema/sieb_ssa_Contact_Contact"
  ...
  <xsd:element name="sieb_ssa_Contact_Contact">
    <xsd:annotation>
      <xsd:appinfo>
        <bx:boDefinition version="1.0.0">
          <bx:appSpecificInfo>ON=Contact;CN=Contact</bx:appSpecificInfo>
        </bx:boDefinition>
      </xsd:appinfo>
    </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Id" minOccurs="0">
        <xsd:annotation>
          <xsd:appinfo>
            <bx:boAttribute>
              <bx:appSpecificInfo>FN=Id</bx:appSpecificInfo>
              <bx:attributeInfo isForeignKey="false"
                isKey="true" isRequired="true" />
            </bx:boAttribute>
          </xsd:appinfo>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="FirstName" minOccurs="1">
        <xsd:annotation>
          <xsd:appinfo>
            <bx:appSpecificInfo>FN=First Name</bx:appSpecificInfo>
            <bx:attributeInfo isForeignKey="false"
              isKey="false" isRequired="false" />
          </xsd:appinfo>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="LastName" minOccurs="1">
        <xsd:annotation>
          <xsd:appinfo>
            <bx:appSpecificInfo>FN=Last Name</bx:appSpecificInfo>
            <bx:attributeInfo isForeignKey="false"
              isKey="false" isRequired="false" />
          </xsd:appinfo>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

---

Abbildung 12. XML-Schemadatei für ein unstrukturiertes Geschäftsobjekt (Teil 1 von 2)

|

---

```

<xsd:element name="AccountId" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:appSpecificInfo>FN=Account Id</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false"
      isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="PrimaryAccountName" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:appSpecificInfo>FN=Primary Account Name</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="PrimaryPostalCode" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:appSpecificInfo>FN=Primary Postal Code</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
</xsd:element>
<xsd:element name="PrimaryStreetAddress" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:appSpecificInfo>FN=Primary Street Address</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="SalesRep" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:boAttribute>
    <bx:appSpecificInfo>FN=Sales Rep</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="State" minOccurs="0">
  <xsd:annotation>
    ...
    <bx:boAttribute>
    <bx:appSpecificInfo>FN=State</bx:appSpecificInfo>
    <bx:attributeInfo isForeignKey="false" isKey="false" />
    ...
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="ObjectEventId"
  type="xsd:string" minOccurs="0" />
</xsd:sequence>
...
</xsd:schema>

```

---

Abbildung 12. XML-Schemadatei für ein unstrukturiertes Geschäftsobjekt (Teil 2 von 2)

Die DB2-Steuerzentrale generiert die nachstehende Anweisung CREATE NICKNAME auf Grundlage der in Abb. 12 auf Seite 159 gezeigten XSD-Datei. Für jede Eingabespalte wird ein Wert für die Option TEMPLATE angegeben. Die Spaltenoptionsschablonen werden der Kurznamenoptionsschablone zugeordnet. Außerdem wird im Wert der Option TEMPLATE auf Kurznamenebene ebenfalls auf die Eingabespalten verwiesen. Die Kurznamenoptionsschablone stellt die Struktur für das Eingabegeschäftsobjekt zur Verfügung. Der Wert des Attributs 'minOccurs' für jeden der Eingabespaltenverweise im Kurznamenschablonenwert legt fest, ob es sich bei der betreffenden Eingabespalte um eine erforderliche oder eine optionale Spalte handelt. Der Verweis für die ID-Spalte wird als &Id[1,1] angegeben. Die Verweise für die Spalten 'FirstName' und 'LastName' werden als &FirstName [0,1] und &LastName [0,1] angegeben. Alle Ausgabespalten umfassen einen Wert für die Spaltenoption XPATH. Der Kurzname stellt ein unstrukturiertes Geschäftsobjekt dar, das keine untergeordneten Elemente (Elemente mit der Kardinalität 'n') enthält:

```
CREATE NICKNAME sieb_ssa_Contact_Contact_NN(
  Id VARCHAR(15)  OPTIONS(XPATH './ns1:Id/text()'),
    TEMPLATE '<ns1:Id>&column</ns1:Id>'),
  FirstName VARCHAR(50)  OPTIONS(XPATH './ns1:FirstName/text()'),
    TEMPLATE '<ns1:FirstName>&column</ns1:FirstName>'),
  LastName VARCHAR(50)  OPTIONS(XPATH './ns1:LastName/text()'),
    TEMPLATE '<ns1:LastName>&column</ns1:LastName>'),
  AccountId VARCHAR(255)
    OPTIONS(XPATH './ns1:AccountId/text()'),
  PrimaryAccountName VARCHAR(100)
    OPTIONS(XPATH './ns1:PrimaryAccountName/text()'),
  PrimaryPostalCode VARCHAR(30)
    OPTIONS(XPATH './ns1:PrimaryPostalCode/text()'),
  PrimaryStreetAddress VARCHAR(200)
    OPTIONS(XPATH './ns1:PrimaryStreetAddress/text()'),
  SalesRep VARCHAR(255)  OPTIONS(XPATH './ns1:SalesRep/text()'),
  State VARCHAR(255)  OPTIONS(XPATH './ns1:State/text()')
  FOR SERVER siebel_server
  OPTIONS(XPATH '//ns1:sieb_ssa_Contact_Contact',
    TEMPLATE '<ns1:sieb_ssa_Contact_Contact>
      &Id[1,1] &FirstName[0,1] &LastName[0,1]
    </ns1:sieb_ssa_Contact_Contact>',
  BUSOBJ_NAME 'sieb_ssa_Contact_Contact',
  NAMESPACE 'ns1="http://www.ibm.com/websphere/
  crossworlds/2002/BOSchema/sieb_ssa_Contact_Contact"');
```

## Beispiel 2: Hierarchisches Geschäftsobjekt

In diesem Beispiel besteht die Geschäftsobjektdefinition aus einer Hierarchie mit zwei Ebenen, die sich aus dem Stammgeschäftsobjekt und zwei untergeordneten Geschäftsobjekten bzw. drei .xsd-Dateien zusammensetzt. Es werden lediglich zwei Kurznamen generiert, um die Hierarchie der Geschäftsobjektdefinition darzustellen. Das untergeordnete Geschäftsobjekt 'sap\_customeraddress' hat die Kardinalität 1, die dadurch angezeigt wird, dass in der Definition des Elements kein Attribut 'maxOccurs' angegeben ist. Alle Spalten von 'sap\_customeraddress' sind im Stammkurznamen (Rootkurznamen) 'sap\_bapi\_customer\_getdetail2\_NN' eingeschlossen. Das untergeordnete Geschäftsobjekt 'sap\_customerbankdetail' hat die Kardinalität 'n', die dadurch angezeigt wird, dass in der Definition des Elements das Attribut 'maxOccurs="unbounded"' angegeben ist. Dieses Geschäftsobjekt wird einem eigenen untergeordneten Kurznamen zugeordnet. Dieser Kurzname lautet 'sap\_bapi\_customer\_getdetail2\_sap\_customerbankdetail\_NN'. Der untergeordnete Kurzname wird mit Hilfe einer besonderen Abhängigkeit zwischen Primärschlüssel und Fremdschlüssel dem Rootkurznamen zugeordnet.

---

```

...
<xsd:element name="sap_bapi_customer_getdetail2">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version="3.0.0" />
    </xsd:appinfo>
  </xsd:annotation>

  ...
  <xsd:element name="COMPANYCODE" minOccurs="0">
    <xsd:annotation>
      <xsd:appinfo>
        <bx:boAttribute>
          <bx:appSpecificInfo>ICOMPANYCODE:</bx:appSpecificInfo>
          <bx:attributeInfo isForeignKey="false" isKey="true" />
        </bx:boAttribute>
      </xsd:appinfo>
    </xsd:annotation>

    ...
  </xsd:element>
  <xsd:element name="CUSTOMERNO" minOccurs="1">
    <xsd:annotation>
      <xsd:appinfo>
        <bx:boAttribute>
          <bx:appSpecificInfo>ICUSTOMERNO:</bx:appSpecificInfo>
          <bx:attributeInfo isForeignKey="false" isKey="false" />
        </bx:boAttribute>
      </xsd:appinfo>
    </xsd:annotation>

    ...
  </xsd:element>

```

---

Abbildung 13. Hierarchisches SAP-Geschäftsobjekt: customer\_getdetail2 (Teil 1 von 2)

|

---

```

<xsd:element name="sap_customeraddress" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ECUSTOMERADDRESS</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
        <bx:childObjectInfo relationship="Containment" version="3.0.0" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
<xsd:element ref="sap_customeraddress:sap_customeraddress" />
</xsd:sequence>

</xsd:complexType>
</xsd:element>
<xsd:element name="sap_customerbankdetail" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>
          ICUSTOMERBANKDETAIL:ECUSTOMERBANKDETAIL
        </bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
        <bx:childObjectInfo relationship="Containment" version="3.0.0" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="sap_customerbankdetail:sap_customerbankdetail"
        minOccurs="unbounded" />
    </xsd:sequence>
    <xsd:attribute name="size" type="xsd:positiveInteger"
      default="1" />
  </xsd:complexType>
</xsd:element>
<xsd:element name="ObjectEventId" type="xsd:string"
  minOccurs="0" />
</xsd:sequence>
...
<xsd:annotation>
  <xsd:appinfo>
    <bx:boVerb>
      <bx:appSpecificInfo>
        bapi.client.Bapi_customer_getdetail2
      </bx:appSpecificInfo>
    </bx:boVerb>
  </xsd:appinfo>
</xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="Update" />
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
...

```

---

Abbildung 13. Hierarchisches SAP-Geschäftsobjekt: customer\_getdetail2 (Teil 2 von 2)

|



---

```

<xsd:element name="sap_customeraddress">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version="3.0.0">
        <bx:appSpecificInfo>:ECUSTOMERADDRESS</bx:appSpecificInfo>
      </bx:boDefinition>
    </xsd:appinfo>
  </xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element name="CUSTOMER" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:ECUSTOMER</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="true" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="NAME" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:ENAME</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="CITY" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:ECITY</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>

```

---

Abbildung 14. Hierarchisches SAP-Geschäftsobjekt: customer\_address (Teil 1 von 2)

|

---

```

<xsd:element name="POSTL_CODE" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:EPOSTL_CODE</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="STREET" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:ESTREET</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name="REGION" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>:EREGION</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
</xsd:sequence>
...

```

---

Abbildung 14. Hierarchisches SAP-Geschäftsobjekt: customer\_address (Teil 2 von 2)

|

---

```

...
<xsd:element name= "sap_customerbankdetail ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version= "3.0.0 ">
        <bx:appSpecificInfo>
          ICUSTOMERBANKDETAIL:ECUSTOMERBANKDETAIL
        </bx:appSpecificInfo>
      </bx:boDefinition>
    </xsd:appinfo>
  </xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element name= "CUSTOMER " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICUSTOMER:ECUSTOMER</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "true " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>

```

---

Abbildung 15. Hierarchisches SAP-Geschäftsobjekt: bank\_detail (Teil 1 von 2)

|

---

```

<xsd:element name= "BANK_KEY " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>IBANK_KEY:EBANK_KEY</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "false " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name= "BANK_ACCT " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>IBANK_ACCT:EBANK_ACCT</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "false " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name= "CTRL_KEY " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICTRL_KEY:ECTRL_KEY</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "false " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
<xsd:element name= "BANK_REF " minOccurs= "0 ">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>IBANK_REF:EBANK_REF</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey= "false " isKey= "false " />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
...

```

---

Abbildung 15. Hierarchisches SAP-Geschäftsobjekt: bank\_detail (Teil 2 von 2)

Die DB2-Steuerzentrale generiert zwei Kurznamen auf Grundlage der drei SAP-XSD-Dateien (vgl. Abb. 16 auf Seite 168 und Abb. 17 auf Seite 169). Aufgrund der XSD-Dateispezifikationen für das Element 'customerno' ist die Spalte 'customerno' in der Schablone auf Kurznamenebene des Kurznamens 'sap\_bapi\_customer\_getdetail2\_NN' als erforderliche Eingabespalte markiert. 'Customerno' ist mit dem Präfix I und dem Attributwert 'minOccurs=1' markiert.

---

```

CREATE NICKNAME sap_bapi_customer_getdetail2_NN(
  CUSTOMER VARCHAR(10)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:CUSTOMER/text()'),
  NAME VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:NAME/text()'),
  CITY VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:CITY/text()'),
  POSTL_CODE VARCHAR(10)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:POSTL_CODE/text()'),
  STREET VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:STREET/text()'),
  REGION VARCHAR(3)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:REGION/text()'),
  NN_PKEY VARCHAR(16) FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'),
  COMPANYCODE VARCHAR(4) OPTIONS(XPATH './ns3:COMPANYCODE/text()',
    TEMPLATE '<ns3:COMPANYCODE>&column</ns3:COMPANYCODE>'),
  CUSTOMERNO VARCHAR(10) OPTIONS(XPATH './ns3:CUSTOMERNO/text()',
    TEMPLATE '<ns3:CUSTOMERNO>&column</ns3:CUSTOMERNO>'),
  ObjectEventId VARCHAR(48) OPTIONS(XPATH './ns3:ObjectEventId/text()'))
  FOR SERVER sap_server
  OPTIONS(XPATH '//ns3:sap_bapi_customer_getdetail2',
  TEMPLATE '<ns3:sap_bapi_customer_getdetail2>
    &sap_bapi_customer_getdetail2_sap_customerbankdetail_NN[0,1]
    &COMPANYCODE[0,1]
    &CUSTOMERNO[1,1]
  </ns3:sap_bapi_customer_getdetail2>',
  BUSOBJ_NAME 'sap_bapi_customer_getdetail2',
  NAMESPACES 'ns1="http://www.ibm.com/websphere/
    crosswor1ds/2002/BOSchema/sap_customeraddress",
    ns2="http://www.ibm.com/websphere/
    crosswor1ds/2002/BOSchema/sap_customerbankdetail",
    ns3="http://www.ibm.com/websphere/
    crosswor1ds/2002/BOSchema/sap_bapi_customer_getdetail2"');

```

---

Abbildung 16. Kurzname 'getdetail2'

I

---

```

CREATE NICKNAME sap_bapi_customer_getdetail2_sap_customerbankdetail_nn(
  CUSTOMER VARCHAR(10)  OPTIONS(XPATH './ns2:CUSTOMER/text()'),
                        TEMPLATE '<ns2:CUSTOMER>&column</ns2:CUSTOMER>'),
  BANK_KEY VARCHAR(15)  OPTIONS(XPATH './ns2:BANK_KEY/text()'),
                        TEMPLATE '<ns2:BANK_KEY>&column</ns2:BANK_KEY>'),
  BANK_ACCT VARCHAR(18) OPTIONS(XPATH './ns2:BANK_ACCT/text()'),
                        TEMPLATE '<ns2:BANK_ACCT>&column</ns2:BANK_ACCT>'),
  CTRL_KEY VARCHAR(2)   OPTIONS(XPATH './ns2:CTRL_KEY/text()'),
                        TEMPLATE '<ns2:CTRL_KEY>&column</ns2:CTRL_KEY>'),
  BANK_REF VARCHAR(20)  OPTIONS(XPATH './ns2:BANK_REF/text()'),
                        TEMPLATE '<ns2:BANK_REF>&column</ns2:BANK_REF>'),
  NN_FKEY VARCHAR(16)  FOR BIT DATA NOT NULL
                        OPTIONS(FOREIGN_KEY 'SAP_BAPI_CUSTOMER_GETDETAIL2_NN'))
FOR SERVER sap_server
OPTIONS(XPATH './ns3:sap_customerbankdetail/ns2:sap_customerbankdetail',
  TEMPLATE '<ns3:sap_customerbankdetail>
<ns2:sap_customerbankdetail>
  &CUSTOMER[0,1]
  &BANK_KEY[0,1]
  &BANK_ACCT[0,1]
  &CTRL_KEY[0,1]
  &BANK_REF[0,1]
</ns2:sap_customerbankdetail>
</ns3:sap_customerbankdetail>',
  NAMESPACE 'ns1="http://www.ibm.com/websphere/
crossworlds/2002/BOSchema/sap_customeraddress",
ns2="http://www.ibm.com/websphere/
crossworlds/2002/BOSchema/sap_customerbankdetail",
ns3="http://www.ibm.com/websphere/
crossworlds/2002/BOSchema/sap_bapi_customer_getdetail2"');

```

---

Abbildung 17. Kurzname 'customerbankdetail'

### Beispiel 3: Primär- und Fremdschlüssel

Die Spalten PRIMARY\_KEY (Primärschlüssel) und FOREIGN\_KEY (Fremdschlüssel) werden verwendet, um die Abhängigkeiten zwischen den übergeordneten und untergeordneten Kurznamen zu definieren. Jeder übergeordnete Kurzname muss über eine Primärschlüsselspaltenoption verfügen. Untergeordnete Kurznamen eines übergeordneten Kurznamens werden mit Hilfe der Fremdschlüsselspaltenoption definiert, die auf die Primärschlüsselspalte eines übergeordneten Kurznamens verweist. Ein Kurzname kann über mehrere untergeordnete Elemente verfügen; er kann jedoch nur ein übergeordnetes Element haben.

Primär- und Fremdschlüsselwerte für den WebSphere Business Integration-Wrapper sind jeweils nur innerhalb einer einzigen Abfrage gültig und eindeutig. Ein Primär- und Fremdschlüssel kann nicht verwendet werden, um eine Zeile mit einer zweiten Abfrage abzurufen. Die Werte können nicht für eine andere Tabelle bestehen bleiben, da die Eindeutigkeit der Werte nicht gewährleistet ist, wenn die betreffende Tabelle mit mehreren Abfragen aufgefüllt wird.

Die folgenden CREATE NICKNAME-Anweisungen leiten sich aus den XML-Schemadefinitionsdateien ab, die in Abb. 13 auf Seite 162, Abb. 14 auf Seite 164 und Abb. 15 auf Seite 166 abgebildet sind. Der Fremdschlüssel 'nn\_fkey' führt zu einer eindeutigen Zuordnung des untergeordneten Kurznamens 'sap\_bapi\_customer\_getdetail2\_sap\_customerbankdetail\_nn' zum übergeordneten Kurznamen 'sap\_bapi\_customer\_getdetail2\_nn'. Der übergeordnete Kurzname verwendet außerdem einen Verweis auf den untergeordneten Kurznamen in der Struktur der Kurznamenoptionsschablone.

```

CREATE NICKNAME sap_bapi_customer_getdetail2_NN(
  CUSTOMER VARCHAR(10)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:CUSTOMER/text()'),
  NAME VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:NAME/text()'),
  ...
  NN__PKEY VARCHAR(16) FOR BIT DATA NOT NULL
    OPTIONS(PRIMARY_KEY 'YES'),
  ...
  TEMPLATE '<ns3:sap_bapi_customer_getdetail2>
    &sap_bapi_customer_getdetail2_sap_customerbankdetail_NN[0,1]
    &COMPANYCODE[0,1]
    &CUSTOMERNO[1,1]
  </ns3:sap_bapi_customer_getdetail2>',
  ...
CREATE NICKNAME sap_bapi_customer_getdetail2_sap_customerbankdetail_NN(
  CUSTOMER VARCHAR(10) OPTIONS(XPATH './ns2:CUSTOMER/text()'),
    TEMPLATE '<ns2:CUSTOMER>&column</ns2:CUSTOMER>'),
  BANK_KEY VARCHAR(15) OPTIONS(XPATH './ns2:BANK_KEY/text()'),
    TEMPLATE '<ns2:BANK_KEY>&column</ns2:BANK_KEY>'),
  BANK_ACCT VARCHAR(18) OPTIONS(XPATH './ns2:BANK_ACCT/text()'),
    TEMPLATE '<ns2:BANK_ACCT>&column</ns2:BANK_ACCT>'),
  CTRL_KEY VARCHAR(2) OPTIONS(XPATH './ns2:CTRL_KEY/text()'),
    TEMPLATE '<ns2:CTRL_KEY>&column</ns2:CTRL_KEY>'),
  BANK_REF VARCHAR(20) OPTIONS(XPATH './ns2:BANK_REF/text()'),
    TEMPLATE '<ns2:BANK_REF>&column</ns2:BANK_REF>'),
  NN__FKEY VARCHAR(16) FOR BIT DATA NOT NULL
    OPTIONS(FOREIGN_KEY 'SAP_BAPI_CUSTOMER_GETDETAIL2_NN'))
  FOR SERVER sap_server
  OPTIONS(XPATH './ns3:sap_customerbankdetail/ns2:sap_customerbankdetail',
  TEMPLATE '<ns3:sap_customerbankdetail>
    <ns2:sap_customerbankdetail>
      &CUSTOMER[0,1]
      &BANK_KEY[0,1]
      &BANK_ACCT[0,1]
      &CTRL_KEY[0,1]
      &BANK_REF[0,1]
    </ns2:sap_customerbankdetail>
  </ns3:sap_customerbankdetail>',
  ...

```

#### Beispiel 4: Verwenden von Namensbereichen zum Auflösen von Präfixen in XPath-Ausdrücken

Die Option NAMESPACES (Namensbereiche) ist eine Liste mit durch Kommata getrennten Name/Wert-Paaren. Diese Option löst die in XPath-Ausdrücken verwendeten Präfixe mit den in den XML-Schemata definierten URIs von Namensbereichen auf. Diese XPath-Ausdrücke werden auf die Geschäftsobjekte (das XML-Dokument) angewendet, die vom WebSphere Business Integration-Adapter zurückgegeben werden. Das folgende Beispiel enthält Namensbereichspräfixe sowie deren Definitionen:

```

CREATE NICKNAME sap_customer
(
  sap_customeraddress_CUSTOMER VARCHAR(10)
    OPTIONS(XPATH './ns5:sap_customeraddress/
      ns2:sap_customeraddress/ns2:CUSTOMER/text()'),
  sap_customeraddress_NAME VARCHAR(35)
    OPTIONS(XPATH './ns5:sap_customeraddress/
      ns2:sap_customeraddress/ns2:NAME/text()'),

```

```

...
sap_bapi_customer_getdet1_PKEY VARCHAR(16) FOR BIT DATA NOT NULL
  OPTIONS(PRIMARY_KEY 'YES'),
COMPANYCODE VARCHAR(4)
  OPTIONS(XPATH './ns5:COMPANYCODE/text()',
    TEMPLATE '<ns5:COMPANYCODE>&column</ns5:COMPANYCODE>'),
CUSTOMERNO VARCHAR(10)
  OPTIONS(XPATH './ns5:CUSTOMERNO/text()',
    TEMPLATE '<ns5:CUSTOMERNO>&column</ns5:CUSTOMERNO>'),
ObjectEventId VARCHAR(48)
  OPTIONS(XPATH './ns5:ObjectEventId/text()')
)
FOR SERVER SAP_SOURCE
  OPTIONS (
    XPATH '//ns5:sap_bapi_customer_getdetail2',
    TEMPLATE
      '<ns5:sap_bapi_customer_getdetail2>
        &customerbankdetail_NN[0,1] &COMPANYCODE[0,1] &CUSTOMERNO[1,1]
      </ns5:sap_bapi_customer_getdetail2>',
    BUSOBJ_NAME 'sap_bapi_customer_getdetail2',
    NAMESPACE '
ns2="http://www.ibm.com/websphere/
  crossworlds/2002/B0Schema/sap_customeraddress",
...
ns5="http://www.ibm.com/websphere/
  crossworlds/2002/B0Schema/sap_bapi_customer_getdetail2",
ns6="http://www.ibm.com/websphere/
  crossworlds/2002/B0Schema/sap_return"
);

```

#### Zugehörige Konzepte:

- „Der WebSphere Business Integration-Wrapper“ auf Seite 137

#### Zugehörige Tasks:

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengesetzter Datenbanken“ auf Seite 144
- „Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen“ auf Seite 148

#### Zugehörige Referenzen:

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172



---

## Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services

### Gleichheitszeichen als Vergleichselement

Als Vergleichselement für Eingabespalten ist ausschließlich das Gleichheitszeichen gültig. Für Ausgabespalten sind alle Vergleichselemente gültig.

Das nachstehende Beispiel gibt eine Fehlermeldung zurück, die darüber informiert, dass das Vergleichselement für die betreffende Spalte nicht unterstützt wird. In diesem Beispiel ist die Spalte `zipcode` eine Eingabespalte:

```
SELECT return FROM gettemp WHERE zipcode<'95141'
```

Das nachstehende Beispiel zeigt eine gültige Abfrage, bei der ein Gleichheitszeichen als Vergleichselement für die Eingabespalte verwendet wird. Der Kurzname `'customers'` wird mit einer lokalen DB2 UDB-Tabelle verknüpft, die Kunden-IDs (`custid`) enthält. Die Abfrage enthält ein zusätzliches Vergleichselement für die Spalte `'Sales'`, bei der es sich ausschließlich um eine Ausgabespalte handelt.

```
SELECT a.name, a.address
       FROM customers a, local_table b
       WHERE
           a.customer_id=b.custid AND
           a.Sales > 300000;
```

### Vergleichselemente für erforderliche Eingabespalten

Sie müssen Werte in Vergleichselementen mit Gleichheitszeichen für alle erforderlichen Eingabespalten in Ihren SQL-Abfragen für die Kurznamenhierarchie, auf die Sie verweisen, angeben. Für alle Abfragen, die gegen diese Einschränkung verstoßen, gibt der Wrapper den `SQLCODE`-Wert 901 zurück.

### Vergleichselemente IN oder OR

Bei WebSphere Business Integration-Wrappern und Web-Services-Wrappern sind für Eingabespalten keine `IN`- oder `OR`-Vergleichselemente zulässig.

Die nachstehenden Beispiele zeigen ungültige Abfragen. Der Kurzname `'customers'` verfügt über eine erforderliche Eingabespalte namens `'customer_id'`:

```
SELECT * FROM customers
       WHERE customer_id IN (12345, 67890, 11223);
SELECT * FROM customers
       WHERE customer_id IN (SELECT custid FROM local_table); )
```

Bei den WebSphere Business Integration-Wrappern können Sie jedoch `IN`-Vergleichselemente für Listen mit erforderlichen Eingabespalten verwenden, wenn Sie einen eindeutigen Index (`UNIQUE INDEX`) mit dem Parameter `SPECIFICATION ONLY` für die erforderlichen Eingabespalten definieren:

```
CREATE UNIQUE INDEX myuindex ON customers(customer_id) SPECIFICATION ONLY;
```

## Verknüpfungen für optionale Eingabespalten

Das nachstehende Beispiel zeigt eine Einschränkung beim Verknüpfen von optionalen Eingabespalten. Sie können keine optionalen Eingabespalten aus einer lokalen Tabelle oder einem lokalen Kurznamen verknüpfen. Wenn das WSDL-Dokument eine Eingabekurznamenspalte als optional generiert und Sie diese Spalte in einer Verknüpfung verwenden wollen, müssen Sie die DDL editieren, um die betreffende Spalte in eine erforderliche Eingabespalte zu ändern.

In diesem Beispiel wird der Kurzname `order` des Web-Services-Wrappers mit der optionalen Eingabespalte `shipping_method` erstellt. Die folgende Anweisung ist eine gültige Abfrage, da sie im Vergleichselement ein Literal verwendet:

```
SELECT * FROM order
  WHERE part="hammer" AND shipping_method="FEDEX";
```

Wenn Sie jedoch eine lokale Tabelle namens `orderparts`, die Ersatzteile (`parts`) und Versandmethoden (`shipping_method`) definiert, in die Abfrage einschließen und die Tabelle eine optionale Spalte namens `shipping_method` enthält, ist die Anweisung ungültig:

```
SELECT * FROM
  order o, orderparts op
 WHERE
  o.part="hammer" AND
  o.shipping_method=op.shipping_method
```

Bei WebSphere Business Integration-Wrappern können Vergleichselemente für optionale Eingabespalten eines Kurznamens unter Umständen im Pushdown-Modus an den WebSphere Business Integration-Adapter gesendet werden. DB2 UDB hat die Möglichkeit, diese Vergleichselemente lokal auf die Zeilen anzuwenden, die aus der Anwendungsdatenquelle abgerufen werden. Um sicherzustellen, dass Vergleichselemente für eine bestimmte Eingabespalte stets im Pushdown-Modus an den Adapter gesendet werden, müssen Sie die Eingabespalte als erforderlich deklarieren. Jede Abfrage für die Kurznamenhierarchie muss Vergleichselementwerte für die erforderlichen Eingabespalten einschließen.

Um sicherzustellen, dass gültige Ergebnisse zurückgegeben werden, muss es sich bei den verknüpften Eingabespalten um erforderliche Spalten für Web-Services-Wrapper handeln.

## Äußere Verknüpfungen

Äußere Verknüpfungen zwischen Kurznamen, bei denen der Primärschlüssel eines übergeordneten Kurznamens und der Fremdschlüssel von untergeordneten Kurznamenspalten verwendet werden, werden nicht unterstützt.

Wenn ein übergeordnetes Element in einem XML-Dokument keine untergeordneten Elemente enthält und Sie eine innere Verknüpfung zwischen dem übergeordneten Kurznamen und dem untergeordneten Kurznamen verwenden, werden für das betreffende Element keine Zeilen zurückgegeben. Sind im SAP-System beispielsweise für einen bestimmten Kunden (`customer`) keine Informationen zur Bankverbindung (`bankdetail`) vorhanden, werden für den betreffenden Kunden keine Zeilen für den Kurznamen `'sap_bapi_customer_getdetail2_sap_customerbankdetail_NN'` zurückgegeben.

Die folgenden CREATE NICKNAME-Anweisungen definieren die Spalten, die in der Beispielabfrage verwendet werden:

```
CREATE NICKNAME sap_bapi_customer_getdetail2_NN(
...
  NAME VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:NAME/text()'),
...
  NN_PKEY VARCHAR(16) OPTIONS(PRIMARY_KEY 'YES'),
  COMPANYCODE VARCHAR(4) OPTIONS(XPATH './ns3:COMPANYCODE/text()',
    TEMPLATE '<ns3:COMPANYCODE>&column</ns3:COMPANYCODE>'),
  CUSTOMERNO VARCHAR(10) OPTIONS(XPATH './ns3:CUSTOMERNO/text()',
    TEMPLATE '<ns3:CUSTOMERNO>&column</ns3:CUSTOMERNO>'),
...
  FOR SERVER sap_server
  OPTIONS(XPATH '//ns3:sap_bapi_customer_getdetail2',
    TEMPLATE '<ns3:sap_bapi_customer_getdetail2>
      &sap_bapi_customer_getdetail2_sap_customerbankdetail_NN[0,1]
      &COMPANYCODE[0,1]
      &CUSTOMERNO[1,1]
    </ns3:sap_bapi_customer_getdetail2>',
...

```

Abbildung 18. Auszug aus dem Kurznamen 'getdetail2'

```
CREATE NICKNAME sap_bapi_customer_getdetail2_sap_customerbankdetail_NN(
  CUSTOMER VARCHAR(10) OPTIONS(XPATH './ns2:CUSTOMER/text()',
    TEMPLATE '<ns2:CUSTOMER>&column</ns2:CUSTOMER>'),
  BANK_KEY VARCHAR(15) OPTIONS(XPATH './ns2:BANK_KEY/text()',
    TEMPLATE '<ns2:BANK_KEY>&column</ns2:BANK_KEY>'),
  BANK_ACCT VARCHAR(18) OPTIONS(XPATH './ns2:BANK_ACCT/text()',
    TEMPLATE '<ns2:BANK_ACCT>&column</ns2:BANK_ACCT>'),
  CTRL_KEY VARCHAR(2) OPTIONS(XPATH './ns2:CTRL_KEY/text()',
    TEMPLATE '<ns2:CTRL_KEY>&column</ns2:CTRL_KEY>'),
  BANK_REF VARCHAR(20) OPTIONS(XPATH './ns2:BANK_REF/text()',
    TEMPLATE '<ns2:BANK_REF>&column</ns2:BANK_REF>'),

  NN_FKEY VARCHAR(16) OPTIONS(FOREIGN_KEY 'SAP_BAPI_CUSTOMER_GETDETAIL2_NN'))
  FOR SERVER sap_server
  OPTIONS(XPATH './ns3:sap_customerbankdetail/ns2:sap_customerbankdetail',
    TEMPLATE '<ns3:sap_customerbankdetail>
      <ns2:sap_customerbankdetail>
        &CUSTOMER[0,1]
        &BANK_KEY[0,1]
        &BANK_ACCT[0,1]
        &CTRL_KEY[0,1]
        &BANK_REF[0,1]
      </ns2:sap_customerbankdetail>
    </ns3:sap_customerbankdetail>',
...

```

Abbildung 19. Auszug aus dem Kurznamen 'customerbankdetail'

In folgendem Beispiel gibt die Abfrage keine Zeilen zurück, da eine Bedingung für eine innere Verknüpfung zwischen den beiden Kurznamen vorhanden ist:

```
SELECT a.name, b.bank_key
FROM sap_bapi_customer_getdetail2_NN a,
     sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
WHERE a.customerno='1234567890'
AND a.NN_PKEY=b.NN_FKEY;
```

Wenn eine Kurznamendefinition eines WebSphere Business Integration-Wrappers oder eines Web-Services-Wrappers erforderliche Eingabespalten enthält, wird keine linke äußere Verknüpfung zwischen diesem Kurznamen und einer anderen lokalen DB2 UDB-Tabelle bzw. anderen Kurznamen unterstützt.

#### Zugehörige Konzepte:

- „Option TEMPLATE auf Kurznamen- und Spaltenebene“ auf Seite 151

#### Zugehörige Tasks:

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengeschlossener Datenbanken“ auf Seite 144
- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 439

#### Zugehörige Referenzen:

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175
- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158
- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463

---

## Geschäftsanwendungsdatenquellen – Beispielabfragen

### Beispiel 1: Verknüpfen von übergeordneten und untergeordneten Kurznamen

Wenn eine SELECT-Anweisung Spalten mit untergeordneten Kurznamen enthält, müssen Sie mit dem übergeordneten Kurznamen ein Verknüpfungsprädikat angeben. Die Verknüpfungsbedingungen sind erforderlich, um die Abhängigkeiten zwischen übergeordneten und untergeordneten Elementen innerhalb der Kurznamenhierarchie beizubehalten. Geben Sie die Bedingungen für die Verknüpfung von Primär- und Fremdschlüsseln für jedes Paar aus über- und untergeordnetem Kurznamen in der Hierarchie an, indem Sie die hierarchische Zuordnung vom untergeordneten Kurznamen, auf den verwiesen wird, zum übergeordneten Kurznamen für die Hierarchie einschließen.

Die folgenden Abfragen sind ungültig, da sie nicht alle Elemente enthalten, die erforderlich sind, um die Kurznamenhierarchie beizubehalten:

```
SELECT * FROM <name_des_untergeordneten_kurznamens>;
SELECT b.col1
FROM <übergeordneter_kurzname> a,<untergeordneter_kurzname> b
WHERE a.required_column=<wert>;
```

Das folgende Beispiel zeigt eine gültige Abfrage, bei der die Kurznamenhierarchie beibehalten wird:

```

| SELECT b.col1, a.col1a
| FROM <übergeordneter_kurzname> a,<untergeordneter_kurzname> b
| WHERE a.primary_key_column=b.foreign_key_column
| AND a.required_column=<wert>;

```

In folgendem Beispiel sind alle erforderlichen übergeordneten und untergeordneten Eingabespalten in den Vergleichselementen in der WHERE-Klausel eingeschlossen. Die WHERE-Klausel umfasst Verknüpfungsprädikate, die eine Gleichberechtigung zwischen der übergeordneten Primärschlüsselspalte und einer untergeordneten Fremdschlüsselspalte angeben:

```

| SELECT a.customer, a.name, b.bank_key, b.bank_acct
| FROM sap_bapi_customer_getdetail2_NN a,
|      sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
| WHERE a.nn_pkey=b.nn_fkey
| AND a.customer = 'ABC'

```

### Beispiel 2: Sichten

Sie haben die Möglichkeit, zwei Typen von Sichten zu erstellen:

- Eine Sicht, die von Spalten in einem untergeordneten Kurznamen abgeleitet wird, sodass Sie Abfragen direkt für den untergeordneten Kurznamen absetzen können, ohne die Bedingungen für die Verknüpfung von über- und untergeordneten Kurznamen in die Abfragen einschließen zu müssen.

```

| CREATE VIEW view1 (
|     customer, bankkey, bankact, customerno )
| AS (SELECT b.customer, b.bank_key, b.bank_acct,
|       a.customerno
| FROM sap_bapi_customer_getdetail2_NN a,
|      sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
| WHERE a.NN_PKEY=b.NN_FKEY);

```

Die Definition der Sicht umfasst die erforderliche Eingabespalte 'customerno' für die Kurznamenhierarchie. Abfragen, die die Sicht 'view1' verwenden, müssen Vergleichselementwerte für die erforderliche Spalte einschließen, wie aus folgendem Beispiel ersichtlich wird:

```

| SELECT * FROM view1
| WHERE customerno='1234567890';

```

- Eine globale Sicht, die alle Spalten der übergeordneten und untergeordneten Kurznamen umfasst, mit Ausnahme der Primärschlüssel- und Fremdschlüsselspalten.

### Beispiel 3: Erforderliche Eingabespalten

Abfragen müssen Vergleichselementwerte für alle erforderlichen Eingabespalten umfassen. Bei den erforderlichen Eingabespalten handelt es sich um die Spalten mit Definitionen für TEMPLATE-Spaltenoptionen und einem Referenzwert von [1,1] im Wert der Kurznamenoption TEMPLATE. Der Wrapper gibt einen Fehler für alle Abfragen zurück, die in den Vergleichselementen nicht die erforderlichen Eingabespalten enthalten.

Das folgende Beispiel zeigt eine ungültige Abfrage. Der Kurzname 'customers' enthält die erforderliche Eingabespalte 'customer\_id', die hier nicht angegeben wird:

```

| SELECT * FROM customers;

```

Das folgende Beispiel zeigt eine gültige Abfrage:

```

| SELECT * FROM customers WHERE customer_id = 123;

```

| Das folgende Beispiel zeigt eine lokale Tabelle in DB2 UDB, die Kunden-IDs (customer\_id) in der Spalte 'custid' der lokalen Tabelle 'local\_table' enthält. Bei diesem Beispiel handelt es sich um eine innere Verknüpfung zwischen dem WebSphere Business Integration-Kurznamen und der lokalen Tabelle:

```
| SELECT a.name, a.address  
|       FROM customers a, local_table b  
|       WHERE a.customer_id=b.custid;
```

| **Zugehörige Konzepte:**

- „Der WebSphere Business Integration-Wrapper“ auf Seite 137

| **Zugehörige Tasks:**

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengeschlossener Datenbanken“ auf Seite 144
- „Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen“ auf Seite 148

| **Zugehörige Referenzen:**

- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172



---

## Kapitel 9. Konfigurieren des Zugriffs auf Datenquellen der DB2-Produktfamilie

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Datenbanken der DB2-Produktfamilie gespeichert sind. Zu diesen Datenbanken gehören:

- DB2 UDB für Linux, UNIX und Windows
- DB2 UDB für z/OS und OS/390
- DB2 UDB für iSeries
- DB2 Server für VM und VSE

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Datenquellen der DB2-Produktfamilie gespeichert sind. Der Zugriff auf Datenquellen der DB2-Produktfamilie kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält eine Liste der Tasks, die Sie ausführen müssen, wenn Sie den Server mit zusammengeschlossenen Datenbanken konfigurieren, zeigt Beispiele zu den benötigten SQL-Anweisungen und bietet Informationen zur Optimierung und Fehlerbehebung beim Konfigurieren des Servers mit zusammengeschlossenen Datenbanken.

---

### Hinzufügen von Datenquellen der DB2-Produktfamilie zu einem Server mit zusammengeschlossenen Datenbanken

#### Hinzufügen von Datenquellen der DB2-Produktfamilie zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen der DB2-Produktfamilie zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken auf Datenquellen der DB2-Produktfamilie mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

#### Voraussetzungen:

- Ein DB2-Server, der für den Zusammenschluss von Datenquellen konfiguriert ist.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.



### **Einschränkungen:**

Sie können keinen Kurznamen für den Aliasnamen einer DB2-Datenquelle erstellen, wenn Sie auf in DB2 für Linux, UNIX und Windows Version 8.1 gespeicherte Daten zugreifen.

### **Vorgehensweise:**

Gehen Sie wie folgt vor, um DB2-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Katalogisieren Sie den Knoten.
2. Katalogisieren Sie die ferne Datenbank.
3. Registrieren Sie den Wrapper.
4. Registrieren Sie die Serverdefinition.
5. Erstellen Sie die Benutzerzuordnungen.
6. Testen Sie die Verbindung zum DB2-Server.
7. Registrieren Sie Kurznamen für DB2-Tabellen und -Sichten.

### **Zugehörige Konzepte:**

- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

### **Zugehörige Tasks:**

- „Überprüfung des Parameters FEDERATED“ auf Seite 50
- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Katalogisieren eines Knoteneintrags in das zusammengeschlossene Knotenverzeichnis“ auf Seite 180
- „Katalogisieren der fernen Datenbank im Datenbankverzeichnis des Systems zusammengeschlossener Datenbanken“ auf Seite 181
- „Registrieren des DB2-Wrappers“ auf Seite 182
- „Registrieren der Serverdefinitionen für eine DB2-Datenquelle“ auf Seite 184
- „Erstellen der Benutzerzuordnung für eine DB2-Datenquelle“ auf Seite 186
- „Testen der Verbindung zum DB2-Datenquellenserver“ auf Seite 188
- „Registrieren von Kurznamen für DB2-Tabellen und -Sichten“ auf Seite 189
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43

### **Zugehörige Referenzen:**

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## **Katalogisieren eines Knoteneintrags in das zusammengeschlossene Knotenverzeichnis**

Das Katalogisieren eines Knoteneintrags in das zusammengeschlossene Knotenverzeichnis ist Teil der umfangreicheren Task des Hinzufügens von Datenquellen der DB2-Produktfamilie zu Servern mit zusammengeschlossenen Datenbanken.

Katalogisieren Sie einen Eintrag im Knotenverzeichnis des Servers mit zusammengeschlossenen Datenbanken, um auf die Speicherposition der DB2-Datenquelle zu zeigen.

Der Server mit zusammengeschlossenen Datenbanken verwendet diesen Eintrag zur Ermittlung der korrekten Zugriffsmethode für die Verbindung mit einer DB2-Datenquelle.

#### Vorgehensweise:

Gehen Sie wie folgt vor, um einen Knoteneintrag in das zusammengeschlossene Knotenverzeichnis zu katalogisieren:

1. Legen Sie das zu verwendende Übertragungsprotokoll fest.
2. Übergeben Sie den entsprechenden Befehl zum Katalogisieren des Knoteneintrags.
  - Wenn Sie als Übertragungsprotokoll Transmission Control Protocol/Internet Protocol (TCP/IP) verwenden, geben Sie den Befehl **CATALOG TCPIP NODE** aus.

Beispiel:

```
CATALOG TCPIP NODE DB2NODE REMOTE SYSTEM42 SERVER DB2TCP42
```

Der Wert *DB2NODE* ist der Name, den Sie zu dem zu katalogisierenden Knoten zuordnen. *REMOTE SYSTEM42* ist der Hostname des Systems, auf dem sich die Datenquelle befindet. *SERVER DB2TCP42* ist der Servicenamen oder die primäre Portnummer des Datenbankmanagerexemplars des Servers. Wenn ein Servicenamen verwendet wird, muss dabei die Groß-/Kleinschreibung beachtet werden.

- Wenn Sie als Übertragungsprotokoll SNA verwenden, geben Sie den Befehl **CATALOG APPC NODE** aus.

Beispiel:

```
CATALOG APPC NODE DB2NODE REMOTE DB2CPIC SECURITY PROGRAM
```

Der Wert *DB2NODE* ist der Name, den Sie zu dem zu katalogisierenden Knoten zuordnen. *REMOTE DB2CPIC* ist der Name der logischen Einheit (Logical Unit, LU) des SNA-Partners im fernen Partnerknoten. *SECURITY PROGRAM* gibt an, dass sowohl ein Benutzername als auch ein Kennwort in die Zuordnungsanforderung aufgenommen werden müssen, die an die Partner-LU gesendet wird.

Die nächste Task in dieser Tasksequenz ist das Katalogisieren der fernen Datenbank im Datenbankverzeichnis des Systems zusammengeschlossener Datenbanken.

#### Zugehörige Tasks:

- „Katalogisieren der fernen Datenbank im Datenbankverzeichnis des Systems zusammengeschlossener Datenbanken“ auf Seite 181

## Katalogisieren der fernen Datenbank im Datenbankverzeichnis des Systems zusammengeschlossener Datenbanken

Das Katalogisieren der fernen Datenbank in das Datenbankverzeichnis des Systems zusammengeschlossener Datenbanken ist Teil der umfangreicheren Task des Hinzufügens von Datenquellen der DB2-Produktfamilie zu Servern mit zusammengeschlossenen Datenbanken.

Durch Katalogisieren der fernen Datenbank in das Systemdatenbankverzeichnis des Servers mit zusammengeschlossenen Datenbanken geben Sie an, zu welcher DB2-Datenquelle der Server mit zusammengeschlossenen Datenbanken eine Verbindung herstellen soll.

### Vorgehensweise:

Gehen Sie wie folgt vor, um die ferne Datenbank in das Systemdatenbankverzeichnis des Servers mit zusammengeschlossenen Datenbanken zu katalogisieren:

1. Verwenden Sie die Funktion 'Clientkonfiguration - Unterstützung' (Client Configuration Assistant = CCA).

Bei Servern mit zusammengeschlossenen Datenbanken unter UNIX können Sie alternativ auch den Befehl **CATALOG DATABASE** verwenden. Beispiel:

```
CATALOG DATABASE DB2DB390 AS CLIENTS390 AT NODE DB2NODE AUTHENTICATION DCS
```

Der Wert *DB2DB390* ist der Name der fernen Datenbank, die Sie in das Systemdatenbankverzeichnis des Servers mit zusammengeschlossenen Datenbanken katalogisieren. *AS CLIENTS390* ist der Aliasname für die Datenbank, die katalogisiert wird. Wenn Sie keinen Aliasnamen angeben, verwendet der Datenbankmanager den Namen der Datenbank (z. B. *DB2DB390*) als Aliasnamen. *AT NODE DB2NODE* ist der Name des Knotens, den Sie beim Katalogisieren des Knoteneintrags im Knotenverzeichnis angegeben haben. *AUTHENTICATION SERVER* gibt an, dass im DB2-Datenquellenknoten die Authentifizierung durchgeführt wird.

2. Ist der Name der fernen Datenbank länger als acht Zeichen, muss durch Ausgeben des Befehls **CATALOG DCS DATABASE** ein Eintrag im DCS-Verzeichnis erstellt werden. Beispiel:

```
CATALOG DCS DATABASE SALES400 AS SALES_DB2DB400
```

Der Wert *SALES400* ist der Aliasname der zu katalogisierenden fernen Datenbank. Dieser Name sollte mit dem Namen eines Eintrags im Systemdatenbankverzeichnis des Servers mit zusammengeschlossenen Datenbanken übereinstimmen, der dem fernen Knoten zugeordnet ist. Dies ist derselbe Name wie der im Befehl **CATALOG DATABASE** eingegebene. *AS SALES\_DB2DB400* ist der Name der Zielhostdatenbank, die Sie katalogisieren möchten.

Die nächste Task in dieser Tasksequenz ist das Registrieren des DB2-Wrappers.

### Zugehörige Tasks:

- „Katalogisieren eines Knoteneintrags in das zusammengeschlossene Knotenverzeichnis“ auf Seite 180
- „Registrieren des DB2-Wrappers“ auf Seite 182

## Registrieren des DB2-Wrappers

Die Registrierung des DB2-Wrappers ist Teil der umfangreicheren Task des Hinzufügens von Datenquellen der DB2-Produktfamilie zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf Datenquellen der DB2-Produktfamilie registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER aus, und geben Sie den Standardnamen für den Wrapper an.

Beispiel:

```
CREATE WRAPPER DRDA
```

**Empfehlung:** Verwenden Sie den Standardnamen für Wrapper, DRDA. Registrieren Sie einen Wrapper mit dem Standardnamen, verwendet der Server mit zusammengeschlossenen Datenbanken automatisch den standardmäßigen Bibliotheksnamen, der diesem Wrapper zugeordnet ist.

Sollte zwischen dem Namen des Wrappers und einem bereits vorhandenen Wrappernamen in der zusammengeschlossenen Datenbank ein Namenskonflikt entstehen, können Sie den Standardnamen für den Wrapper durch einen beliebigen Namen ersetzen. Verwenden Sie einen anderen Namen als den Standardnamen, müssen Sie den Parameter LIBRARY in die Anweisung CREATE WRAPPER mit einschließen.

Um beispielsweise einen Wrapper mit dem Namen db2\_wrapper auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER db2_wrapper LIBRARY 'libdb2drda.a'
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung CREATE WRAPPER finden Sie in der Liste der Bibliotheksdateien von DB2-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den DB2-Wrapper.

### Zugehörige Tasks:

- „Registrieren der Serverdefinitionen für eine DB2-Datenquelle“ auf Seite 184

### Zugehörige Referenzen:

- „Bibliotheksdateien des DB2-Wrappers“ auf Seite 183
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien des DB2-Wrappers

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den DB2-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, werden 3 Bibliotheksdateien zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden folgende Wrapperbibliotheksdateien zum Verzeichnispfad hinzugefügt: libdb2drda.a, libdb2drdaF.a und libdb2drdaU.a.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

Tabelle 34. Positionen und Dateinamen von DB2-Wrapperbibliotheken

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2drda.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2drda.sl
Linux	/opt/IBM/db2/V8.1/lib	libdb2drda.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2drda.so
Windows	%DB2PATH%\bin	db2drda.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist C:\Program Files\IBM\SQLLIB.

#### Zugehörige Tasks:

- „Registrieren des DB2-Wrappers“ auf Seite 182

## Registrieren der Serverdefinitionen für eine DB2-Datenquelle

Die Registrierung der Serverdefinitionen für eine DB2-Datenquelle ist Teil der umfangreicheren Task des Hinzufügens von Datenquellen der DB2-Produktfamilie zu Servern mit zusammengeschlossenen Datenbanken.

In der zusammengeschlossenen Datenbank müssen Sie jeden DB2-Server definieren, auf den Sie zugreifen möchten. Wenn Sie die Serverdefinition registrieren, stellt der Server mit zusammengeschlossenen Datenbanken eine Verbindung zum DB2-Server her und bindet Pakete an die Datenbank. Da die Informationen für die Berechtigung und das Kennwort nicht im globalen Katalog der zusammengeschlossenen Datenbanken gespeichert werden, müssen Sie sie in die Serverdefinition mit aufnehmen.

#### Vorgehensweise:

Gehen Sie wie folgt vor, um eine Serverdefinition für eine DB2-Datenquelle zu registrieren.

Beispiel:

```
CREATE SERVER server_name TYPE DB2/ZOS VERSION 6 WRAPPER DRDA
  AUTHORIZATION "name1" PASSWORD "passwd1"
  OPTIONS (DBNAME 'db_name')
```

Der Name, den Sie zu einem Server zuordnen, muss eindeutig sein. Doppelte Servernamen sind nicht zulässig.

Die von Ihnen angegebene Option VERSION ist die Version der DB2-Datenbank, auf die Sie zugreifen möchten.

Der Name des Parameters WRAPPER muss der Name sein, den Sie in der Anweisung CREATE WRAPPER angegeben haben.

Obwohl der Datenbankname in der Anweisung CREATE SERVER als Option angegeben wird, ist er für DB2-Datenquellen erforderlich.

Wenn Sie die Anweisung CREATE SERVER ausgeben, testet der Server mit zusammengeschlossenen Datenbanken die Verbindung zum DB2-Datenquellenserver.

Nachdem Sie die Serverdefinition registriert haben, können Sie Serveroptionen durch Ausgeben der Anweisung ALTER SERVER hinzufügen oder löschen.

Die nächste Task in dieser Tasksequenz ist das Erstellen der Benutzerzuordnung für eine DB2-Datenquelle.

#### Zugehörige Tasks:

- „Hinzufügen von Datenquellen der DB2-Produktfamilie zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 179
- „Erstellen der Benutzerzuordnung für eine DB2-Datenquelle“ auf Seite 186

#### Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- „Unterstützte Datenquellen“ auf Seite 6
- „Anweisung CREATE SERVER - Beispiele für den DB2-Wrapper“ auf Seite 185

## Anweisung CREATE SERVER - Beispiele für den DB2-Wrapper

In diesem Abschnitt sind mehrere Beispiele für die Verwendung der Anweisung CREATE SERVER zur Registrierung von Servern für Wrapper für Datenquellen der DB2-Produktfamilie enthalten. Der Abschnitt enthält ein vollständiges Beispiel für die Erstellung eines Servers mit allen erforderlichen Parametern sowie ein Beispiel zur Erstellung mit zusätzlichen Serveroptionen.

#### Vollständiges Beispiel:

Das nachfolgende Beispiel zeigt die Erstellung einer Serverdefinition für einen DB2-Wrapper unter Verwendung der Anweisung CREATE SERVER:

```
CREATE SERVER DB2SERVER TYPE DB2/ZOS VERSION 6 WRAPPER DRDA
  AUTHORIZATION "spalten" PASSWORD "db2guru"
  OPTIONS (DBNAME 'CLIENTS390')
```

#### DB2SERVER

Ein Name, den Sie dem DB2-Datenbankserver zuordnen. Dieser Name muss eindeutig sein. Doppelte Servernamen sind nicht zulässig.

#### TYPE DB2/ZOS

Gibt den Typ des Datenquellenservers an, für den Sie den Zugriff konfigurieren.

#### VERSION 6

Die Version des DB2-Datenbankservers, auf den Sie zugreifen möchten.

#### WRAPPER DRDA

Der Name, den Sie in der Anweisung CREATE WRAPPER angegeben haben.

#### AUTHORIZATION "spalten"

Die Berechtigungs-ID in der Datenquelle. Diese ID muss in der Datenquelle über die Berechtigung BINDADD verfügen. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

#### PASSWORD "db2guru"

Das Kennwort, das in der Datenquelle der Berechtigungs-ID zugeordnet ist. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

#### DBNAME 'CLIENTS390'

Der Aliasname für die DB2-Datenbank, auf die Sie zugreifen möchten. Die-

sen Aliasnamen haben Sie beim Katalogisieren der Datenbank mit dem Befehl **CATALOG DATABASE** definiert. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

Dieser Datenbankname ist für DB2-Datenquellen erforderlich.

### Beispiel mit Serveroptionen:

Wenn Sie die Serverdefinition registrieren, können Sie in der Anweisung **CREATE SERVER** zusätzliche Serveroptionen angeben. Diese Optionen beinhalten allgemeine Serveroptionen und spezifische Serveroptionen für DB2-Datenquellen.

Das nachfolgende Beispiel zeigt eine Serverdefinition mit der Option **CPU\_RATIO**.

```
CREATE SERVER DB2SERVER TYPE DB2/ZOS VERSION 6 WRAPPER DRDA
  AUTHORIZATION "spalten" PASSWORD "db2guru"
  OPTIONS (DBNAME 'CLIENTS390', CPU_RATIO '0.001')
```

Setzen Sie die Option **CPU\_RATIO** auf '0.001', wird dadurch angegeben, dass die CPU der fernen Datenquelle im Vergleich zum Server mit zusammengeschlossenen Datenbanken über die 1000-fache freie Kapazität verfügt.

### Zugehörige Tasks:

- „Registrieren der Serverdefinitionen für eine DB2-Datenquelle“ auf Seite 184

### Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang C, „Gültige Servertypen in SQL-Anweisungen“, auf Seite 635

## Erstellen der Benutzerzuordnung für eine DB2-Datenquelle

Das Erstellen der Benutzerzuordnung für eine DB2-Datenquelle ist Teil der umfangreicheren Task des Hinzufügens von Datenquellen der DB2-Produktfamilie zu Servern mit zusammengeschlossenen Datenbanken. Wird versucht, auf einen DB2-Server zuzugreifen, stellt der Server mit zusammengeschlossenen Datenbanken unter Verwendung einer gültigen Benutzer-ID mit Kennwort eine Verbindung zur Datenquelle her. Sie müssen eine Zuordnung zwischen der Benutzer-ID des Servers mit zusammengeschlossenen Datenbanken mit zugehörigem Kennwort und der entsprechenden Benutzer-ID mit Kennwort für die Datenquelle definieren. Diese Zuordnung muss für alle Benutzer-IDs erstellt werden, die auf das System zusammengeschlossener Datenbanken zum Senden verteilter Anforderungen zugreifen.

### Vorgehensweise:

Geben Sie eine Anweisung **CREATE USER MAPPING** aus, um die lokale Benutzer-ID zur Benutzer-ID und dem zugehörigen Kennwort des DB2-Servers mit zusammengeschlossenen Datenbanken zuzuordnen.

Beispiel:

```
CREATE USER MAPPING FOR USERID SERVER DB2SERVER
  OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

**REMOTE\_AUTHID** ist die Verbindungsberechtigungs-ID, nicht die Bindeberechtigungs-ID.



Die nächste Task in dieser Tasksequenz ist das Testen der Verbindung zum DB2-Datenquellenserver.

**Zugehörige Tasks:**

- „Testen der Verbindung zum DB2-Datenquellenserver“ auf Seite 188

**Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE USER MAPPING - Beispiele für den DB2-Wrapper“ auf Seite 187

## Anweisung CREATE USER MAPPING - Beispiele für den DB2-Wrapper

Dieser Abschnitt enthält Beispiele für die Verwendung der Anweisung CREATE USER MAPPING zur Zuordnung einer lokalen Benutzer-ID zu der Benutzer-ID mit Kennwort des DB2-Servers. Dieser Abschnitt enthält ein vollständiges Beispiel mit allen erforderlichen Parametern sowie ein Beispiel für die Verwendung des DB2-Sonderregisters USER zusammen mit der Anweisung CREATE USER MAPPING.

**Vollständiges Beispiel:**

Das folgende Beispiel zeigt die Zuordnung einer lokalen Benutzer-ID zur Benutzer-ID des DB2-Servers:

```
CREATE USER MAPPING FOR DB2USER SERVER DB2SERVER
    OPTIONS (REMOTE_AUTHID 'db2admin', REMOTE_PASSWORD 'day2night')
```

**DB2USER**

Gibt die lokale Benutzer-ID an, die Sie zu einer Benutzer-ID zuordnen, die auf dem Server einer Datenquelle der DB2-Produktfamilie definiert ist.

**SERVER DB2SERVER**

Gibt den Namen des Datenquellenservers der DB2-Produktfamilie an, den Sie in der Anweisung CREATE SERVER definiert haben.

**REMOTE\_AUTHID 'db2admin'**

Gibt die Benutzer-ID für die Verbindungsberechtigung auf dem Datenquellenserver der DB2-Produktfamilie an, zu der Sie DB2USER zuordnen. Verwenden Sie einfache Anführungszeichen, um die Groß-/Kleinschreibung dieses Wertes zu erhalten, sofern Sie in der Anweisung CREATE SERVER nicht die Serveroption FOLD\_ID auf 'U' oder 'L' gesetzt haben.

**REMOTE\_PASSWORD 'day2night'**

Gibt das Kennwort an, das 'db2admin' zugeordnet ist. Verwenden Sie einfache Anführungszeichen, um die Groß-/Kleinschreibung dieses Wertes zu erhalten, sofern Sie in der Anweisung CREATE SERVER nicht die Serveroption FOLD\_PW auf 'U' oder 'L' gesetzt haben.

**Beispiel mit Sonderregister:**

Nachfolgend ist ein Beispiel für die Anweisung CREATE USER MAPPING aufgeführt, bei dem das Sonderregister USER enthalten ist:

```
CREATE USER MAPPING FOR USER SERVER DB2SERVER
    OPTIONS (REMOTE_AUTHID 'db2admin', REMOTE_PASSWORD 'day2night')
```



Sie können das DB2-Sonderregister USER dazu verwenden, die Berechtigungs-ID der Person, die die Anweisung CREATE USER MAPPING ausgibt, zur Berechtigungs-ID der Datenquelle zuzuordnen, die in der Benutzeroption REMOTE\_AUTHID angegeben wurde.

**Zugehörige Tasks:**

- „Erstellen der Benutzerzuordnung für eine DB2-Datenquelle“ auf Seite 186

**Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

## Testen der Verbindung zum DB2-Datenquellenserver

Das Testen der Verbindung zum DB2-Datenquellenserver ist Teil der umfangreicheren Task des Hinzufügens von Datenquellen der DB2-Produktfamilie zu Servern mit zusammengeschlossenen Datenbanken.

Sie können die Verbindung zum DB2-Server durch Verwendung der von Ihnen festgelegten Serverdefinition und Benutzerzuordnungen testen.

**Vorgehensweise:**

Gehen Sie wie folgt vor, um die Verbindung zu testen:

1. Eröffnen Sie eine Durchgriffssitzung zum Ausgeben einer Anweisung SELECT für die DB2-Systemtabellen.

Beispiel:

- Für DB2 für z/OS und OS/390:

```
SET PASSTHRU server_name
SELECT count(*) FROM sysibm.systables
SET PASSTHRU RESET
```

- Für DB2 für iSeries:

```
SET PASSTHRU remote_server_name
SELECT count(*) FROM qsys2.systables
SET PASSTHRU RESET
```

Wenn die Anweisung SELECT eine Anzahl zurückgibt, sind Ihre Serverdefinition und Ihre Benutzerzuordnung ordnungsgemäß eingerichtet.

2. Gibt die Anweisung SELECT einen Fehler zurück, sind eventuell folgende Schritte erforderlich:
  - Den fernen Server prüfen, um sicherzustellen, dass er gestartet wurde.
  - Die Listenerfunktion auf dem fernen Server prüfen, um sicherzustellen, dass sie für ankommende Verbindungsanforderungen konfiguriert wurde.
  - Ihre Benutzerzuordnung prüfen, um sicherzustellen, dass die Einstellungen für die Optionen REMOTE\_AUTHID und REMOTE\_PASSWORD für Verbindungen zum DB2-Server gültig sind.
  - Die DB2-Katalogeinträge für den Knoten und die Datenbank prüfen.
  - Die Einstellungen Ihrer Variablen für den DB2-Server mit zusammengeschlossenen Datenbanken prüfen, um sicherzustellen, dass Sie auf den fernen DB2-Server zugreifen können. Zu diesen Variablen gehören die Systemumgebungsvariablen, die db2dj.ini-Variablen und die Variable DB2COMM der DB2-Profilregistrierdatenbank (db2set).
  - Ihre Serverdefinition überprüfen. Falls erforderlich, löschen Sie die Serverdefinition und erstellen Sie sie erneut.

- Ihre Benutzerzuordnung überprüfen. Falls erforderlich, ändern Sie die Benutzerzuordnung oder erstellen Sie eine neue Benutzerzuordnung.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für DB2-Tabellen und -Sichten.

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für DB2-Tabellen und -Sichten“ auf Seite 189
- „Festlegen der Umgebungsvariablen für eine Datenquelle“ auf Seite 67

**Zugehörige Referenzen:**

- „ALTER USER MAPPING statement“ in *SQL Reference, Volume 2*

## Registrieren von Kurznamen für DB2-Tabellen und -Sichten

Die Registrierung der Kurznamen für DB2-Tabellen und -Sichten ist Teil der umfangreicheren Task des Hinzufügens von Datenquellen der DB2-Produktfamilie zu einem Server mit zusammengeschlossenen Datenbanken.

Die zusammengeschlossene Datenbank verwendet Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Diese Statistiken werden gesammelt, wenn Sie unter Verwendung der Anweisung CREATE NICKNAME einen Kurznamen für ein Datenquellenobjekt registrieren. Die zusammengeschlossene Datenbank stellt das Vorhandensein des Objekts in der Datenquelle sicher und versucht dann, statistische Daten zu bestehenden Datenquellen zu sammeln. Informationen, die für das Optimierungsprogramm nützlich sind, werden aus den Katalogen der Datenquellen gelesen und in den globalen Katalog auf dem Server mit zusammengeschlossenen Datenbanken aufgenommen. Da einige oder alle der Kataloginformationen der Datenquelle vom Optimierungsprogramm verwendet werden können, ist es ratsam, Statistiken (unter Verwendung des Datenquellenbefehls, der RUNSTATS entspricht) in der Datenquelle zu aktualisieren, bevor ein Kurzname erstellt wird.

Verwenden Sie die Anweisung CREATE NICKNAME, um für eine Sicht oder Tabelle, die sich in Ihrer Datenquelle der DB2-Produktfamilie befindet, einen Kurznamen zu registrieren. Verwenden Sie diese Kurznamen anstatt der Namen des Datenquellenobjekts, wenn Sie die Datenquelle der DB2-Produktfamilie abfragen.

**Einschränkungen:**

Sie können für einen DB2-Aliasnamen keinen Kurznamen erstellen.

**Vorgehensweise:**

Übergeben Sie zum Registrieren eines Kurznamens die Anweisung CREATE NICKNAME.

Beispiel:

```
CREATE NICKNAME DB2NICKNAME FOR
DB2SERVER.remote_schema.remote_table
```

Kurznamen können bis zu 128 Zeichen lang sein.

Wiederholen Sie diesen Schritt für jede DB2-Tabelle, für die Sie einen Kurznamen erstellen möchten.

Wenn Sie den Kurznamen registrieren, verwendet der Server mit zusammenge- schlossenen Datenbanken die Verbindung dazu, den Datenquellenkatalog abzufr- agen. Diese Abfrage testet Ihre Verbindung zur Datenquelle unter Verwendung des Kurznamens. Kann die Verbindung nicht hergestellt werden, empfangen Sie eine Fehlernachricht.

**Zugehörige Referenzen:**

- „RUNSTATS Command“ in *Command Reference*
- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den DB2-Wrapper“ auf Seite 190

## Anweisung CREATE NICKNAME - Beispiele für den DB2-Wrapper

Dieser Abschnitt enthält ein Beispiel für die Verwendung der Anweisung CREATE NICKNAME zur Registrierung eines Kurznamens für eine DB2-Tabelle oder -Sicht, auf die Sie zugreifen möchten.

Das folgende Beispiel zeigt eine Anweisung CREATE NICKNAME:

```
CREATE NICKNAME DB2SALES FOR DB2SERVER.SALESDATA.EUROPE
```

*DB2SALES*

Ein eindeutiger Kurzname, der zur Identifizierung der DB2-Tabelle oder -Sicht verwendet wird.

**Anmerkung:** Der Kurzname besteht aus zwei Teilen - dem Schema und dem Kurznamen. Wenn Sie bei der Registrierung des Kurznamens das Schema auslassen, wird die Berechtigungs-ID des Benutzers, der den Kurznamen erstellt, zum Schema des Kurznamens.

*DB2SERVER.SALESDATA.EUROPE*

Eine dreiteilige Kennung für das ferne Objekt:

- *DB2SERVER* ist der Name, den Sie dem DB2-Datenbankserver in der Anweisung CREATE SERVER zugeordnet haben.
- *SALESDATA* ist der Name des fernen Schemas, zu dem die Tabelle oder Sicht gehört. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.
- *EUROPE* ist der Name der fernen Tabelle oder Sicht, auf die Sie zugreifen möchten.

**Zugehörige Tasks:**

- „Ändern eines Kurznamens“ auf Seite 587
- „Registrieren von Kurznamen für DB2-Tabellen und -Sichten“ auf Seite 189

**Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

---

## Kapitel 10. Konfigurieren des Zugriffs auf Documentum-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Documentum-Datenquellen gespeichert sind. Der Zugriff auf Documentum-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält folgende Informationen:

- Erläuterungen zu Documentum.
- Liste der auszuführenden Tasks.
- Beispiele zu den benötigten SQL-Anweisungen.
- Liste der dem Documentum-Wrapper zugeordneten Fehlernachrichten.

---

### Was ist Documentum?

Documentum ist eine Dokumentverwaltungssoftware, die Funktionen zur Verwaltung von Dokumentinhalten sowie Attribute für die Rückgabe (Check-In), die Entnahme (Check-Out), den Arbeitsablauf und die Versionsverwaltung bereit stellt. Documentum ist ein Client-/Serversystem mit drei Ebenen, das auf einer relationalen Datenbank aufbaut.

Eine Docbase (Dokumentdatenbank) ist ein Documentum-Repository, in dem Dokumentinhalte, Attribute, Beziehungen, Versionen, Ausgabeformate, Formate, Arbeitsabläufe und Sicherheitsinformationen gespeichert werden. Documentum Query Language (DQL), eine SQL-Variante, wird zum Abfragen von Documentum-Daten verwendet. Eine Docbase ist das Äquivalent zu einem Oracle-Exemplar oder einer DB2<sup>®</sup>-Datenbank mit Dokumentinhaltsdateien. Die Metadaten werden im zu Grunde liegenden Verwaltungssystem für relationale Datenbanken (RDBMS) gespeichert, der Inhalt als große Binärobjekte (BLOBs) in der Datenbank bzw. als Dateien im Dateisystem des Serversystems. Weitere Informationen zu Documentum finden Sie in den Documentum-Handbüchern.

Der Wrapper für Documentum ermöglicht das Hinzufügen einer Documentum-Datenquelle zu einem DB2-System mit einer zusammengeschlossenen Datenbank. Wenn Sie die Documentum-Datenquelle zu einem System mit einer zusammengeschlossenen Datenbank hinzufügen, können Sie mit Hilfe von SQL-Anweisungen auf Objekte und registrierte Tabellen in einer Documentum-Docbase zugreifen und diese abfragen. Sie können diese Daten in andere Datenquellen innerhalb des Systems mit einer zusammengeschlossenen Datenbank integrieren, ohne die Daten aus der ursprünglichen Datenquelle verschieben zu müssen. Der Documentum-Wrapper verwendet eine Client-Bibliothek als Schnittstelle zum Documentum-Server. Der Documentum-Wrapper bietet Zugriff auf zwei Versionen des Documentum-Servers: EDMS 98 (auch als Version 3 bezeichnet) und 4i. In Abb. 20 auf Seite 192 ist die Funktionsweise des Documentum-Wrappers dargestellt.

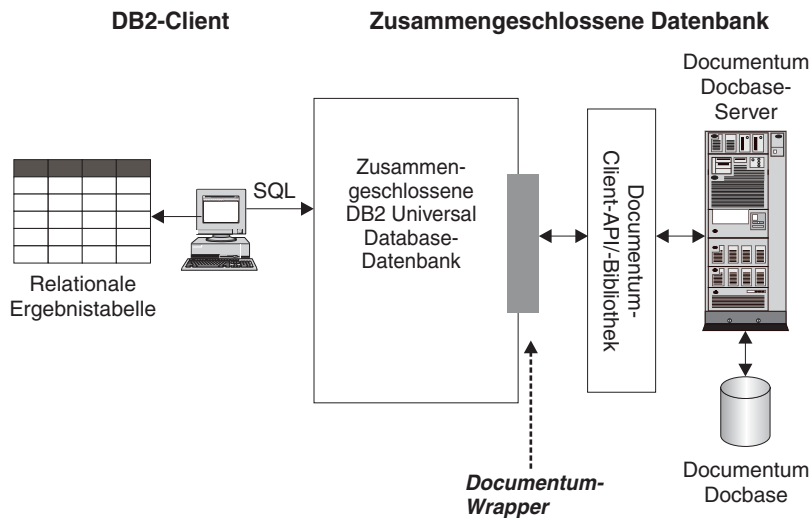


Abbildung 20. Funktionsweise des Documentum-Wrappers

Nach der Registrierung des Documentum-Wrappers können Documentum-Docbase-Objekte und registrierte Tabellen als relationale Tabellen zugeordnet werden. Hierzu werden Docbase-Attribute Spaltennamen in einer relationalen DB2-Tabelle zugeordnet.

Beispiel: In Tabelle 35 ist eine Untermenge von Attributen für den Standarddokumenttyp der Documentum-Docbase - dm\_document - zusammen mit den zugehörigen Daten aufgeführt. Sie entscheiden, dass diese Attributuntermenge für Sie von Bedeutung ist, und möchten für diese Attribute eine Verbindung zu Ihrem System mit einer zusammenschlossenen Datenbank herstellen. Sie nennen diese Datenuntermenge DrugAB\_data.

Tabelle 35. DrugAB\_data

Title	Subject	Authors	Keywords
The effect of drug A on rabbits	Drug A	Curran, L.	rabbits, drug A
Toxicity results for drug A	Drug A	Abelite, P., McMurtrey, K.	toxicity, drug A
Drug B interactions	Drug B	DeNiro, R., Stone, S.	interactions, drug B
Chemical structure of drug B	Drug B	Boyslim, F.	structure, drug B

Nach der Registrierung des Documentum-Wrappers können die Daten mit Hilfe von SQL-Anweisungen abgefragt werden.

In der folgenden Abfrage werden die Titel (Titles) und Autoren (Authors) angezeigt, deren Thema (Subject) das Medikament A (Drug A) ist. Die Ergebnistabelle ist in Tabelle 36 auf Seite 193 dargestellt.

```
SELECT title, authors
FROM drugAB_data
WHERE subject = 'Drug A'
```

Tabelle 36. Abfrageergebnisse

Title	Authors
The effect of drug A on rabbits	Curran, L.
Toxicity results for drug A	Abelite, P., McMurtrey, K.

**Zugehörige Tasks:**

- „Hinzufügen von Documentum-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 193

---

## Hinzufügen von Documentum zu einem Server mit zusammengesetzten Datenbanken

### Hinzufügen von Documentum-Datenquellen zu einem Server mit zusammengesetzten Datenbanken

Um den Server mit zusammengesetzten Datenbanken für den Zugriff auf Documentum-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengesetzten Datenbanken auf Documentum-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengesetzten Datenbanken führt.

**Voraussetzungen:**

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengesetzten Datenbanken fungiert.
- Eine zusammengesetzte Datenbank muss sich auf dem Server mit zusammengesetzten Datenbanken befinden.

**Vorgehensweise:**

Führen Sie folgende Schritte aus, um Documentum-Datenquellen zu einem Server mit zusammengesetzten Datenbanken hinzuzufügen:

1. Stellen Sie die Documentum-Clientbibliothek für den Wrapper bereit.
2. Legen Sie die Documentum-Umgebungsvariablen fest.
3. Registrieren Sie den Wrapper.
4. Registrieren Sie die Serverdefinition.
5. Erstellen Sie die Benutzerzuordnungen.
6. Registrieren Sie die Kurznamen für Documentum Docbase-Objekte und registrierte Tabellen.
7. Registrieren Sie angepasste Funktionen für den Documentum-Wrapper.

**Zugehörige Konzepte:**

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzübersicht“ auf Seite 63

### Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Bereitstellen der Documentum-Clientbibliothek für den Wrapper“ auf Seite 194
- „Festlegen der Documentum-Umgebungsvariablen“ auf Seite 195
- „Registrieren des Documentum-Wrappers“ auf Seite 196
- „Registrieren des Servers für Documentum-Datenquellen“ auf Seite 198
- „Registrieren der Benutzerzuordnungen für Documentum-Datenquellen“ auf Seite 198
- „Registrieren von Kurznamen für Documentum-Datenquellen“ auf Seite 199
- „Registrieren der angepassten Funktionen für den Documentum-Wrapper“ auf Seite 206
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43

### Zugehörige Referenzen:

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## Bereitstellen der Documentum-Clientbibliothek für den Wrapper

Das Bereitstellen der Documentum-Clientbibliothek für den Wrapper ist Teil der umfangreicheren Task des Hinzufügens von Documentum zu einem System zusammengeschlossener Datenbanken.

Um die fehlerfreie Funktion des Wrappers zu gewährleisten, muss die Clientbibliothek für den Wrapper verfügbar sein.

### Voraussetzungen:

Der Documentum-Wrapper verwendet Version 3.1.7a der Clientbibliothek. Wenn Sie Documentum 4i verwenden, benötigen Sie die ältere Version der Clientbibliothek von Documentum (falls diese nicht bereits installiert ist).

### Vorgehensweise:

Um die Documentum-Clientbibliothek für den Wrapper bereitzustellen, erstellen Sie einen symbolischen Link oder kopieren Sie die Clientbibliothek in das entsprechende Verzeichnis auf dem Server mit zusammengeschlossenen Datenbanken. In der folgenden Tabelle werden die Verzeichnisse aufgelistet, in die die Bibliothek kopiert werden sollte.

*Tabelle 37. Clientbibliothek und Kopieren in Verzeichnis geordnet nach Betriebssystem*

Betriebssystem des Servers mit zusammengeschlossenen Datenbanken	Clientbibliothek	Kopieren in Verzeichnis
AIX	libdmcl.a	sqllib/lib
Solaris	libdmcl.so	sqllib/lib
Windows	dmcl32.dll	x:\sqllib\bin



Die nächste Task in dieser Tasksequenz ist das Festlegen der Umgebungsvariablen für den Documentum-Wrapper.

**Zugehörige Tasks:**

- „Festlegen der Documentum-Umgebungsvariablen“ auf Seite 195

## Festlegen der Documentum-Umgebungsvariablen

Das Festlegen der Documentum-Umgebungsvariablen ist ein Teil der umfangreicheren Task des Hinzufügens von Documentum zu einem Server mit zusammengesetzten Datenbanken.

Der Zugriff auf Documentum Docbases wird über die Documentum-Clientdatei `dmcl.ini` gesteuert. Die Umgebungsvariablen des Exemplars der zusammengesetzten Datenbank müssen auf die Documentum-Clientdatei `dmcl.ini` gesetzt sein, um Zugriff auf eine Documentum Docbase zu erhalten.

Gültige Umgebungsvariablen für Documentum lauten wie folgt:

- DOCUMENTUM
- DMCL\_CONFIG

**Voraussetzungen:**

Ein Server mit zusammengesetzten Datenbanken, der für den Zugriff auf Ihre Datenquellen korrekt konfiguriert ist. Dies beinhaltet die Installation und Konfiguration von erforderlicher Software, z. B. der Client-Software der Datenquelle.

**Einschränkungen:**

Siehe hierzu den Abschnitt: Einschränkungen für die Datei `db2dj.ini`

**Vorgehensweise:**

Um die Umgebungsvariablen zu definieren, gehen Sie wie folgt vor:

1. Bearbeiten Sie die Datei `db2dj.ini`, und legen Sie die Umgebungsvariable `DOCUMENTUM` oder `DMCL_CONFIG` fest.
  - Auf Servern mit zusammengesetzten Datenbanken, die unter AIX bzw. Solaris ausgeführt werden, befindet sich die Datei `db2dj.ini` unter `$HOME/sql11ib/cfg`.
  - Auf Servern mit zusammengesetzten Datenbanken, die unter Windows ausgeführt werden, befindet sich die Datei `db2dj.ini` unter `x:\sql11ib\cfg`, wobei `x`: das Laufwerk darstellt, auf dem sich das Verzeichnis `sql11ib` befindet.

Folgende Beispiele zeigen die Syntax für diese Variablen unter UNIX-Betriebssystemen:

`DOCUMENTUM=pfad`

oder

`DMCL_CONFIG=pfad/dmcl.ini`

Dabei ist `pfad` das vollständig qualifizierte Verzeichnis, das die zu verwendende Datei `dmcl.ini` enthält. Weitere Informationen zum Pfad der Datei `dmcl.ini` enthält die mit Documentum mitgelieferte Dokumentation.



| Wenn beide dieser Variablen in der Datei db2dj.ini gesetzt sind, wird der Wert  
| für die Variable DMCL\_CONFIG verwendet. Wenn Sie keine der Variablen in  
| der Datei db2dj.ini setzen, wird ein Fehler zurückgegeben.

2. Stellen Sie sicher, dass der Name eines Dokumentbrokers (docbroker), an den alle für den Zugriff verfügbaren Docbases für das DB2-Exemplar berichten, in der Datei dmcl.ini, wie in Abb. 21 gezeigt, angegeben wird.

---

```
##### DOCUMENTUM CLIENT CONFIGURATION FILE #####  
#  
# Copyright Documentum 1994.  
# Version 3.1 of the Documentum Server.  
#  
# A generated client init file for the Documentum Server.  
#  
# The only REQUIRED information in this file is the  
# [DOCBROKER_PRIMARY] section and an entry for host.  
# The host value should be the name of host on which  
# your network wide DocBroker is running  
  
[DOCBROKER_PRIMARY]  
host = server16.comp2.big.com
```

---

Abbildung 21. Beispiel für die Datei 'dmcl.ini' mit Angabe des Namens eines Dokumentbrokers

3. Um sicherzustellen, dass die Umgebungsvariablen für den Server mit zusammengeschlossenen Datenbanken gesetzt sind, beenden Sie das DB2-Exemplar, und starten Sie es erneut. Geben Sie die folgenden Befehle aus, um das DB2-Exemplar erneut zu starten:

```
db2stop  
db2start
```

Die nächste Task in dieser Tasksequenz ist das Registrieren des Documentum-Wrappers.

#### Zugehörige Tasks:

- „Registrieren des Documentum-Wrappers“ auf Seite 196

#### Zugehörige Referenzen:

- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68

## Registrieren des Documentum-Wrappers

Die Registrierung des Documentum-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von Documentum-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf Documentum-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Bibliotheksdatei des Wrappers aus.

Um beispielsweise einen Wrapper mit dem Namen dctm\_wrapper auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER dctm_wrapper LIBRARY 'libdb2lsdctm.a';
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung CREATE WRAPPER finden Sie in der Liste der Bibliotheksdateien von Documentum-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinition für Documentum-Datenquellen.

### Zugehörige Referenzen:

- „Bibliotheksdateien für den Documentum-Wrapper“ auf Seite 197
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien für den Documentum-Wrapper

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den Documentum-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, werden 3 Bibliotheksdateien zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden folgende Wrapperbibliotheksdateien zum Verzeichnispfad hinzugefügt: libdb2lsdctm.a, libdb2lsdctmF.a und libdb2lsdctmU.a.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 38. Positionen und Dateinamen von Documentum-Wrapperbibliotheken*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2lsdctm.a
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lsdctm.so
Windows	%DB2PATH%\bin	db2lsdctm.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist C:\Program Files\IBM\SQLLIB.

### Zugehörige Tasks:

- „Registrieren des Documentum-Wrappers“ auf Seite 196

## Registrieren des Servers für Documentum-Datenquellen

Die Registrierung der Serverdefinition für Documentum-Datenquellen ist Teil der umfangreicheren Task des Hinzufügens von Documentum zu einem System zusammengesetzter Datenbanken.

### Einschränkungen:

Alle Server, die für dasselbe DB2-Exemplar ausgeführt werden, müssen dieselben Konfigurationsparameter in der Datei `dmcl.ini` verwenden.

### Vorgehensweise:

Sie können eine Serverdefinition von der DB2-Steuerzentrale aus oder über die DB2-Befehlszeile registrieren:

- Um diese Task von der DB2-Steuerzentrale aus auszuführen, verwenden Sie den Assistenten für zusammengesetzte Objekte oder klicken Sie mit der rechten Maustaste den Ordner **Serverdefinitionen** an, und klicken Sie anschließend **Erstellen** an. Das Tool zum Aufspüren (Discover-Tool) ruft die Knotennamen für die Documentum-Server ab. Sie müssen die Informationen für die Serveroptionen `RDBMS_TYPE` und `OS_TYPE` angeben, um die Serverdefinition zu registrieren.
- Um diese Task von der DB2-Befehlszeile aus auszuführen, verwenden Sie die Anweisung `CREATE SERVER`.

Um beispielsweise die Serverdefinition `Dctm_Server1` für einen Documentum-Server zu registrieren, der eine Docbase enthält, die unter AIX ausgeführt wird und Daten mit Hilfe von Oracle speichert, verwenden Sie folgende Anweisung:

```
CREATE SERVER Dctm_Server1
  TYPE DCTM
  VERSION 3
  WRAPPER Dctm_Wrapper
  OPTIONS( NODE 'Dctm_Docbase',
          OS_TYPE 'AIX',
          RDBMS_TYPE 'ORACLE');
```

Die nächste Task in dieser Tasksequenz ist das Erstellen der Benutzerzuordnungen.

### Zugehörige Tasks:

- „Registrieren der Benutzerzuordnungen für Documentum-Datenquellen“ auf Seite 198

### Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- „CREATE SERVER-Anweisungsargumente und -optionen - Documentum-Wrapper“ auf Seite 604

## Registrieren der Benutzerzuordnungen für Documentum-Datenquellen

Das Registrieren von Benutzerzuordnungen für Documentum-Datenquellen ist ein Teil der umfangreicheren Task des Hinzufügens von Documentum zu einem Server mit zusammengesetzten Datenbanken.

Wenn Sie versuchen, auf eine Documentum-Datenquelle zuzugreifen, stellt der Server mit zusammengeschlossenen Datenbanken eine Verbindung zu der Datenquelle her und verwendet dabei eine Benutzer-ID und ein Kennwort, die für die betreffende Datenquelle gültig sind. Sie müssen eine Zuordnung zwischen der Benutzer-ID des Servers mit zusammengeschlossenen Datenbanken mit zugehörigem Kennwort und der entsprechenden Benutzer-ID mit Kennwort für die Datenquelle definieren. Diese Zuordnung muss für alle Benutzer-IDs erstellt werden, die auf das System zusammengeschlossener Datenbanken zum Senden verteilter Anforderungen zugreifen.

#### **Vorgehensweise:**

Um eine Benutzerzuordnung zu erstellen, verwenden Sie die Anweisung CREATE USER MAPPING.

Mit der folgenden Anweisung CREATE USER MAPPING beispielsweise wird der Benutzer Chuck dem Benutzer Charles auf dem Server Dctm\_Server1 zugeordnet.

```
CREATE USER MAPPING FOR Chuck SERVER Dctm_Server1
OPTIONS(REMOTE_AUTHID 'Charles', REMOTE_PASSWORD 'Charles_pw');
```

Sie können das DB2-Sonderregister USER dazu verwenden, die Berechtigungs-ID der Person, die die Anweisung CREATE USER MAPPING ausgibt, zur Berechtigungs-ID der Datenquelle zuzuordnen, die in der Benutzeroption REMOTE\_AUTHID angegeben wurde. Beispiel:

```
CREATE USER MAPPING FOR USER SERVER Dctm_Server1
OPTIONS(REMOTE_AUTHID 'Lisa', REMOTE_PASSWORD 'Lisa_pw');
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für Documentum-Datenquellen.

#### **Zugehörige Tasks:**

- „Registrieren von Kurznamen für Documentum-Datenquellen“ auf Seite 199

#### **Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „CREATE USER MAPPING-Anweisungsoptionen - Documentum-Wrapper“ auf Seite 605

## **Registrieren von Kurznamen für Documentum-Datenquellen**

### **Registrieren von Kurznamen für Documentum-Datenquellen**

Das Registrieren der Kurznamen für Documentum-Datenquellen ist Teil der umfangreicheren Task des Hinzufügens von Documentum zu einem Server mit zusammengeschlossenen Datenbanken.

Nach der Erstellung der Benutzerzuordnungen müssen Sie die Kurznamen registrieren. Registrieren Sie für jeden Documentum-Server, den Sie definieren, jeweils einen Kurznamen für jeden Documentum Docbase-Objekttyp und jede registrierte Tabelle, auf den bzw. die Sie zugreifen möchten. Verwenden Sie diese Kurznamen anstelle der Namen der Docbase, wenn Sie die Documentum-Datenquellen abfragen.

### Vorgehensweise:

Verwenden Sie zum Registrieren von Kurznamen die Anweisung CREATE NICKNAME.

Sie können als Teil der Anweisung CREATE NICKNAME Pseudospalten definieren. Definieren Sie die Pseudospalten nicht, werden sie vom Wrapper erstellt.

Die nächste Task in dieser Tasksequenz ist das Registrieren der angepassten Funktionen für Documentum-Datenquellen.

### Zugehörige Tasks:

- „Registrieren der angepassten Funktionen für den Documentum-Wrapper“ auf Seite 206
- „Einrichten von TurboBlast zum Arbeiten mit dem BLAST-Wrapper“ auf Seite 130

### Zugehörige Referenzen:

- „Pseudospalten“ auf Seite 200
- „CREATE NICKNAME-Anweisungssyntax - Documentum-Wrapper“ auf Seite 606
- „Anweisung CREATE NICKNAME - Beispiel für Documentum-Wrapper“ auf Seite 204

## Pseudospalten

Die mit Documentum verwendete Anweisung CREATE NICKNAME definiert 6 Pseudospalten. Diese Spalten werden für den Zugriff auf Objektivhalte und andere Informationen verwendet.

Tabelle 39 enthält die Pseudospalten und ihre jeweilige Definition.

*Tabelle 39. Namen und Definitionen von Pseudospalten.*

Pseudospaltenname	Definition
GET_FILE	VARCHAR (nnn) <sup>1</sup>
GET_FILE_DEL	VARCHAR (nnn) <sup>1</sup>
GET_RENDITION	VARCHAR (nnn) <sup>1</sup>
GET_RENDITION_DEL	VARCHAR (nnn) <sup>1</sup>
HITS	INTEGER
SCORE	DOUBLE

Hinweis:

1. Die Länge der Variablen VARCHAR richtet sich nach der maximalen Pfadlänge, die für das Betriebssystem des Servers mit zusammengeschlossenen Datenbanken zulässig ist. Auf UNIX-Servern mit zusammengeschlossenen Datenbanken beträgt diese Länge 1024. Auf Windows-Servern mit zusammengeschlossenen Datenbanken beträgt diese Länge 260.

Tabelle 40 enthält die Pseudospalten für SELECT-Klauseln.

*Tabelle 40. Pseudospalten für SELECT-Klauseln*

<b>Pseudospaltenname</b>	<b>Beschreibung</b>
GET_FILE	<p>Ruft die Inhaltsdatei für die aktuelle Zeile zusätzlich zu den Spaltenwerten ab.</p> <p>Die Erweiterung für die Inhaltsdatei ist ihr Documentum-Formatname. Ist eine Datei mit demselben Namen bereits vorhanden, wird sie überschrieben.</p> <p>GET_FILE versucht, das Basisformat des Objekts abzurufen. Sein Wert in der Zeile besteht aus dem vollständig qualifizierten Dateinamen dieser Datei oder aus der Zeichenfolge "no_content".</p> <p>Beispiel:</p> <pre>SELECT object_name, get_file FROM ...</pre> <p>Die Inhaltsdatei wird in das Serververzeichnis gestellt, das durch die Option CONTENT_DIR des Servers angegeben wird. Darüber hinaus wird sie in ein Unterverzeichnis gestellt, das mit Ihrem lokalen DB2-Namen benannt ist. Wenn das Unterverzeichnis nicht vorhanden ist, wird es erstellt.</p> <p>Als Erweiterung wird die DOS-Erweiterung der Datei verwendet, die in der Docbase für den Formattyp des Dokuments definiert ist. So wird beispielsweise ".doc" für MS Word-Dokumente verwendet.</p>
GET_FILE_DEL	<p>Diese Funktion entspricht der Funktion GET_FILE, mit der Ausnahme, dass GET_FILE_DEL zunächst die für die vorhergehende Zeile dieser Abfrage abgerufene Datei (sofern vorhanden) löscht. Sein Wert in der Zeile besteht aus dem vollständig qualifizierten Dateinamen dieser Datei oder aus der Zeichenfolge "no_content".</p>

Tabelle 40. Pseudospalten für SELECT-Klauseln (Forts.)

Pseudospaltenname	Beschreibung
GET_RENDITION	<p>Ruft die Inhaltsdatei dieses Ausgabeformats (Rendition), d. h. einer Kopie des ursprünglichen Dokuments in einem anderen Format, für die aktuelle Zeile zusätzlich zu den Spaltenwerten ab.</p> <p>Die Erweiterung für die Inhaltsdatei ist ihr Documentum-Formatname. Ist eine Datei mit demselben Namen bereits vorhanden, wird sie überschrieben.</p> <p>Um das Ausgabeformat anzugeben, muss in der WHERE-Klausel ein Vergleichselement im Format <code>DCTM.RENDITION_FORMAT(&lt;format&gt;) = 1</code> angegeben werden.</p> <p>Beispiel:</p> <pre>SELECT object_name, get_rendition FROM ...WHERE DCTM.RENDITION_FORMAT('pdf')=1</pre> <p>GET_RENDITION versucht, das genannte Ausgabeformat des Objekts abzurufen. Sein Wert in der Zeile besteht aus dem vollständig qualifizierten Dateinamen dieser Datei oder aus der Zeichenfolge "no_content".</p> <p>Die Inhaltsdatei wird in das Serververzeichnis gestellt, das durch die Option <code>CONTENT_DIR</code> des Servers angegeben wird. Darüber hinaus wird sie in ein Unterverzeichnis gestellt, das mit Ihrem lokalen DB2-Namen benannt ist. Wenn das Unterverzeichnis nicht vorhanden ist, wird es erstellt.</p> <p>Als Erweiterung wird die DOS-Erweiterung der Datei verwendet, die in der Docbase für den Formattyp des Dokuments definiert ist. So wird beispielsweise ".doc" für MS Word-Dokumente verwendet.</p>
GET_RENDITION_DEL	<p>Diese Funktion entspricht der Funktion GET_RENDITION, mit der Ausnahme, dass GET_RENDITION_DEL zunächst die für die vorhergehende Zeile dieser Abfrage abgerufene Datei (sofern vorhanden) löscht. Sein Wert in der Zeile besteht aus dem vollständig qualifizierten Dateinamen dieser Datei oder aus der Zeichenfolge "no_content".</p>

Tabelle 41 auf Seite 203 enthält die Pseudospalten für SELECT-Klauseln in Abfragen mit Suchklauseln.

Tabelle 41. Pseudospalten für SELECT-Klauseln in Abfragen mit Suchklauseln

Pseudospalte	Beschreibung
HITS	<p>Enthält eine ganze Zahl, die die Anzahl der Stellen innerhalb des Dokuments angibt, an denen Übereinstimmungen mit den Suchkriterien gefunden wurden.</p> <p>Beispiel:</p> <pre>SELECT r_object_id, object_name, hits FROM std_doc WHERE DCTM.SEARCH_WORDS (''workflow'' OR ''flowchart'')=1</pre> <p>Für jedes zurückgegebene Dokument wird die Häufigkeit der Wörter "workflow" und "flowchart" im Dokumentinhalt addiert und als HITS-Wert zurückgegeben.</p> <p>Die Pseudospalte HITS empfiehlt sich, wenn die Dokumente nur über eine Inhaltsdatei verfügen. Dies ist normalerweise der Fall. Diese Pseudospalte kann in der Qualifizierung einer WHERE-Klausel für eine SELECT-Anweisung verwendet werden. Sie muss jedoch auch in der SELECT-Klausel angegeben werden.</p>
SCORE	<p>Enthält den Relevanzwert des Dokuments.</p> <p>Verwenden Sie diese Pseudospalte zusammen mit dem Konzeptoperator ACCRUE von Documentum. Beide geben eine Zahl zurück, die angibt, wie viele der angegebenen Wörter in jedem zurückgegebenen Dokument gefunden wurden.</p> <p>Beispiel:</p> <pre>SELECT object_name, score FROM std_doc WHERE DCTM.SEARCH_TOPIC('&lt;ACCRUE&gt;("document","management","workflow")')=1 AND SCORE &gt;=75</pre> <p>Die Anweisung gibt alle Dokumente zurück, in deren Inhalt zwei oder drei der angegebenen Wörter vorhanden sind. Ist in einem Dokument nur eines der Wörter enthalten, wird diesem der Wert 50 zugeordnet. Das bedeutet, das Dokument erfüllt die Kriterien der WHERE-Klausel nicht und wird nicht zurückgegeben. Werden zwei der drei Wörter gefunden, erhält das Dokument den Wert 75. Werden alle drei Wörter gefunden, erhält das Dokument den Wert 88.</p> <p>Die Pseudospalte SCORE wird für Dokumente verwendet, die über eine einzige Inhaltsdatei verfügen. Dies ist normalerweise der Fall.</p> <p>SCORE kann nur dann in einer SELECT-Klausel enthalten sein, wenn WHERE eine der Funktionen SEARCH_WORDS() oder SEARCH_TOPIC() enthält. In einer WHERE-Klausel wird sie zusammen mit dem Konzeptoperator ACCRUE verwendet.</p> <p>Informationen zum Konzeptoperator ACCRUE finden Sie in der Dokumentation zu Documentum.</p>

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für Documentum-Datenquellen“ auf Seite 199
- „Registrieren der angepassten Funktionen für den Documentum-Wrapper“ auf Seite 206



## Anweisung CREATE NICKNAME - Beispiel für Documentum-Wrapper

Mit der folgenden Anweisung CREATE NICKNAME wird der Kurzname `std_doc` definiert. `Std_doc` wird einer Documentum-Docbase mit dem Objekttyp `dm_document` zugeordnet. In Tabelle 42 werden die Documentum-Attribute und -Datentypen relationalen DB2-Spaltennamen und -Datentypen zugeordnet, die dann für die Generierung der Anweisung CREATE NICKNAME verwendet werden.

Tabelle 42. Zuordnung der Documentum-Attribute zu DB2-Spalten für den Kurznamen `std_doc`

Documentum-Attributname	Documentum-Datentyp	DB2-Spaltenname	DB2-Datentyp	Wiederholen?	Nullwerte zulässig?
<code>object_name</code>	<code>string(255)</code>	<code>object_name</code>	<code>varchar</code>	Nein	Nein
<code>r_object_id</code>	<code>ID</code>	<code>object_id</code>	<code>char(16)</code>	Nein	Nein
<code>r_object_type</code>	<code>string(32)</code>	<code>object_type</code>	<code>varchar</code>	Nein	Nein
<code>title</code>	<code>string(255)</code>	<code>title</code>	<code>varchar</code>	Nein	Nein
<code>subject</code>	<code>string(128)</code>	<code>subject</code>	<code>varchar</code>	Nein	Nein
<code>authors</code>	<code>string(32)</code>	<code>author</code>	<code>varchar</code>	Ja	Ja
<code>keywords</code>	<code>string(32)</code>	<code>keyword</code>	<code>varchar</code>	Ja	Ja
<code>r_creation_date</code>	<code>time</code>	<code>creation_date</code>	<code>timestamp</code>	Nein	Ja
<code>r_modify_date</code>	<code>time</code>	<code>modified_date</code>	<code>timestamp</code>	Nein	Ja
<code>a_status</code>	<code>string(16)</code>	<code>status</code>	<code>varchar</code>	Nein	Nein
<code>a_content_type</code>	<code>string(32)</code>	<code>content_type</code>	<code>varchar</code>	Nein	Nein
<code>r_content_size</code>	<code>double</code>	<code>content_size</code>	<code>integer</code>	Nein	Nein
<code>owner_name</code>	<code>string(32)</code>	<code>owner_name</code>	<code>varchar</code>	Nein	Ja

Tabelle 43 enthält jeweils eine Beschreibung aller im Kurznamen verwendeten Documentum-Attribute.

Tabelle 43. Beschreibung der Documentum-Attribute für den Kurznamen `'std_doc'`

Documentum-Attributname	Beschreibung
<code>object_name</code>	Der benutzerdefinierte Name des Objekts.
<code>r_object_id</code>	Die eindeutige, bei der Erstellung definierte Objektkennung für dieses Objekt.
<code>r_object_type</code>	Der bei der Erstellung des Objekts definierte Objekttyp.
<code>title</code>	Der benutzerdefinierte Titel des Objekts.
<code>subject</code>	Das benutzerdefinierte Thema des Objekts.
<code>authors</code>	Die benutzerdefinierte Liste der Autoren für das Objekt.
<code>keywords</code>	Die Liste der benutzerdefinierten Schlüsselwörter für das Objekt.
<code>r_creation_date</code>	Datum und Uhrzeit der Erstellung des Objekts.
<code>r_modify_date</code>	Datum und Uhrzeit der letzten Änderung des Objekts.
<code>a_status</code>	Wird vom Server bei der Weiterleitung einer Router-Task definiert. Der Wert stammt aus den Werten, die <code>attached_task_status</code> im Router-Objekt zugeordnet sind.

Tabelle 43. Beschreibung der Documentum-Attribute für den Kurznamen 'std\_doc' (Forts.)

Documentum-Attributname	Beschreibung
a_content_type	Das Dateiformat des Objektinhalts.
r_content_size	Die Bytezahl des Inhalts. Bei mehrseitigen Dokumenten gibt dieses Attribut die Größe des ersten dem Dokument zugeordneten Inhalts an.
owner_name	Der Name des Objekteigners (der Benutzer, der das Objekt erstellt hat).

Aus Tabelle 42 auf Seite 204 wird die folgende Anweisung CREATE NICKNAME generiert.

```
CREATE NICKNAME std_doc (
  object_name varchar(255) not null,
  object_id char(16) not null OPTIONS(REMOTE_NAME 'r_object_id'),
  object_type varchar(32) not null OPTIONS(REMOTE_NAME 'r_object_type'),
  title varchar(255) not null,
  subject varchar(128) not null,
  author varchar(32) OPTIONS(REMOTE_NAME 'authors', IS_REPEATING 'Y'),
  keyword varchar(32) OPTIONS(REMOTE_NAME 'keywords', IS_REPEATING 'Y'),
  creation_date timestamp OPTIONS(REMOTE_NAME 'r_creation_date'),
  modified_date timestamp OPTIONS(REMOTE_NAME 'r_modify_date'),
  status varchar(16) not null OPTIONS(REMOTE_NAME 'a_status'),
  content_type varchar(32) not null OPTIONS(REMOTE_NAME 'a_content_type'),
  content_size integer not null OPTIONS(REMOTE_NAME 'r_content_size'),
  owner_name varchar(32))
FOR SERVER Dctm_Server2 OPTIONS (REMOTE_OBJECT 'dm_document', IS_REG_TABLE 'N')
```

Nach der Übergabe der Anweisung CREATE NICKNAME können Sie den Kurznamen *std\_doc* für Abfragen Ihres Systems zusammenschlossener Datenbanken verwenden. Darüber hinaus können Sie den Kurznamen *std\_doc* mit anderen Kurznamen und Tabellen im System zusammenschlossener Datenbanken verknüpfen.

Im Katalog ist die Anzahl der Spalten für diesen Kurznamen um 6 höher als in der Anweisung CREATE NICKNAME angegeben. Dies ist auf die Pseudospalten zurückzuführen.

Sie können das Dienstprogramm 'CreateNicknameFile' verwenden, um Documentum-Typen automatisch DB2-Typen zuzuordnen und eine erste Anweisung CREATE NICKNAME zu erstellen.

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für Documentum-Datenquellen“ auf Seite 199

#### Zugehörige Referenzen:

- „CREATE NICKNAME-Anweisungssyntax - Documentum-Wrapper“ auf Seite 606

### Doppelte Definition von Wiederholungsattributen (Documentum-Wrapper)

Um die Abfrageleistung des Wrappers zu optimieren, muss jedes Attribut als der tatsächliche, diesem Attribut äquivalente DB2-Datentyp definiert werden. Das bedeutet, dass der Documentum-Datentyp 'Integer' als DB2-Datentyp 'Integer' definiert werden muss, etc. Diese Definitionen verhindern jedoch die Rückgabe mehrerer Werte für Wiederholungsattribute, die nicht dem Typ VARCHAR entsprechen. Für diese Spalten wird nur der letzte Wert zurückgegeben.

Diese Einschränkung ist vorhanden, da der Wrapper nur eine Ergebniszeile pro Docbase-Objekt zurückgibt, wo immer dies möglich ist. Diese Einschränkung ist nur dann von Bedeutung, wenn Wiederholungsattribute ausgewählt werden. Sie können jedoch eine zweite Spalte für dasselbe ferne Wiederholungsattribut, jedoch mit dem Datentyp VARCHAR definieren.

Dieser Spaltenname wird in der SELECT-Liste verwendet, um alle Werte als eine durch Begrenzer unterteilte Liste der entsprechenden Werte zurückzugeben. (Die Option DELIMITER für die einzelnen Spalten gibt den zu verwendenden Begrenzer an.)

Die lokalen Namen der Spalten mit mehreren Werten sollten standardisiert werden. Hierzu wird das Präfix "m\_" zum lokalen Namen der Spalte hinzugefügt, die als tatsächlicher Datentyp definiert ist.

Beispiel: Angenommen, eine Kurznamenspalte eines Documentum-Wiederholungsattributs hat den Namen approval\_dates und ist mit dem Datentyp TIMESTAMP definiert. Sie können eine zweite Kurznamenspalte mit dem Namen m\_approval\_dates erstellen und diese mit dem Datentyp VARCHAR definieren. Sie können m\_approval\_dates dann in einer SELECT-Liste verwenden, um alle Genehmigungsdaten in einer durch Begrenzer unterteilten Liste zurückzugeben.

Für Wiederholungsattribute, deren tatsächlicher Datentyp VARCHAR lautet, ist keine doppelte Definition erforderlich.

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Documentum-Datenquellen“ auf Seite 199
- „Ändern eines Kurznamens“ auf Seite 587

---

## Abfragen und angepasste Funktionen für Documentum-Datenquellen

### Registrieren der angepassten Funktionen für den Documentum-Wrapper

Das Registrieren von angepassten Funktionen für Documentum-Datenquellen ist Teil der umfangreicheren Task des Hinzufügens von Documentum zu einem System zusammenschlossener Datenbanken. Verwenden Sie die Anweisung CREATE FUNCTION, um eine Reihe von angepassten Funktionen zu registrieren. Sie können diese Funktionen verwenden, um auf einige besondere Funktionen von Documentum, wie beispielsweise die Volltextsuche und das Abrufen von Dokumentinhalten innerhalb von Abfragen, zuzugreifen.

Angepasste Funktionen für Vergleichselemente sind in Tabelle 44 auf Seite 208 aufgeführt.

Verweise auf die Funktion TOPIC beziehen sich auf die Documentum-Funktion, die als Teil des Fremdanbieter-Volltextindexierungssystems von Verity, Inc. bereitgestellt wird.

### Einschränkungen:

Da DB2 den Booleschen Typ nicht unterstützt, müssen die meisten angepassten Funktionen (mit Ausnahme von USER), die in der WHERE-Klausel verwendet werden, eine Überprüfung auf "=1" durchführen, da diese Funktionen für die Rückgabe einer ganzen Zahl definiert sind.

Beispiel:

```
"... WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1"
```

### Vorgehensweise:

Um angepasste Funktionen zu registrieren, verwenden Sie die Anweisung CREATE FUNCTION.

Alle angepassten Funktionen müssen mit dem Schemanamen DCTM registriert werden. Der vollständig qualifizierte Name der jeweiligen Funktion ist DCTM.funktionsname.

Im folgenden Beispiel wird die angepasste Funktion ANY\_EQ registriert.

```
CREATE FUNCTION DCTM.ANY_EQ (CHAR(), CHAR()) RETURNS INTEGER  
AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION
```

Jede angepasste Funktion muss einmal für jede zusammengeschlossene Datenbank registriert werden, für die der Documentum-Wrapper installiert ist.

Die Beispieldatei create\_function\_mappings.ddl im Verzeichnis sqllib/samples/lifesci/dctm unterstützt Sie bei der Registrierung von angepassten Funktionen. Diese Datei enthält Definitionen für die einzelnen angepassten Funktionen. Sie können diese DDL-Datei ausführen, um die angepassten Funktionen für jede zusammengeschlossene Datenbank zu registrieren, für die der Documentum-Wrapper installiert ist.

### Regeln für Zeichenfolgeargumente bei angepassten Funktionen

Alle Argumente, die als Zeichenfolgen übergeben werden, müssen den folgenden Regeln entsprechen:

- Jede Zeichenfolge muss in einfache Anführungszeichen eingeschlossen sein.
- Einfache Anführungszeichen innerhalb von Zeichenfolgen müssen durch zwei einfache Anführungszeichen dargestellt werden.

### Verwenden von angepassten Funktionen in Abfragen

Im folgenden Beispiel ist die Verwendung der angepassten Funktionen in Abfragen dargestellt.

Anzeigen des Objektnamens und des Autors vom Kurznamen std\_doc für Dokumente, die einen oder mehrere Autoren mit dem Namen 'Dave Winters' aufweisen:

```
SELECT object_name,authors FROM std_doc  
WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1
```

Anzeigen des Objektnamens und des Autors vom Kurznamen std\_doc für Dokumente, die einen oder mehrere Autoren mit dem Namen 'Dave Winters' oder 'Jon Doe' aufweisen:

```
SELECT object_name,authors FROM std_doc  
WHERE DCTM.ANY_IN(authors,'Dave Winters','Jon Doe')=1
```

Anzeigen des Objektnamens und Anzeigen von `r_object_id` sowie Abrufen der Inhaltsdatei vom Kurznamen `std_doc` für Dokumente, die Zeichenfolgen ähnlich wie 'Dave Win%' in der Autorenspalte enthalten:

```
SELECT object_name, r_object_id, get_file FROM std_doc
WHERE DCTM.ANY_LIKE(authors,'Dave Win%')=1
```

## Tabelle der angepassten Funktionen

Tabelle 44 enthält eine Aufstellung der angepassten Funktionen für Vergleichselemente.

*Tabelle 44. Angepasste Funktionen für Vergleichselemente*

<b>Funktionsname</b>	<b>Beschreibung</b>
ANY_EQ(arg1, arg2)	<p>Überprüft ein Wiederholungsattribut auf alle Werte, die dem angegebenen Wert entsprechen. Zwei Argumente sind erforderlich:</p> <p><b>arg1</b> Gibt den Namen einer Spalte an, die ein Wiederholungsattribut darstellt.</p> <p><b>arg2</b> Gibt den zu vergleichenden Wert an.</p> <p>Beispiel: ... WHERE DCTM.ANY_EQ(authors,'Dave Winters')=1</p>
ANY_NE(arg1, arg2)	<p>Überprüft ein Wiederholungsattribut auf alle Werte, die dem angegebenen Wert nicht entsprechen. Zwei Argumente sind erforderlich:</p> <p><b>arg1</b> Gibt den Namen einer Spalte an, die ein Wiederholungsattribut darstellt.</p> <p><b>arg2</b> Gibt den zu vergleichenden Wert an.</p> <p>Beispiel: ... WHERE DCTM.ANY_NE(authors,'Dave Winters')=1</p>
ANY_LT(arg1, arg2)	<p>Überprüft ein Wiederholungsattribut auf alle Werte, die kleiner sind als der angegebene Wert. Zwei Argumente sind erforderlich:</p> <p><b>arg1</b> Gibt den Namen einer Spalte an, die ein Wiederholungsattribut darstellt.</p> <p><b>arg2</b> Gibt den zu vergleichenden Wert an.</p> <p>Beispiel: ... WHERE DCTM.ANY_LT(num_approvers,4)=1</p>
ANY_GT(arg1, arg2)	<p>Überprüft ein Wiederholungsattribut auf alle Werte, die größer sind als der angegebene Wert. Zwei Argumente sind erforderlich:</p> <p><b>arg1</b> Gibt den Namen einer Spalte an, die ein Wiederholungsattribut darstellt.</p> <p><b>arg2</b> Gibt den zu vergleichenden Wert an.</p> <p>Beispiel: ... WHERE DCTM.ANY_GT(num_approvers,3)=1</p>

Tabelle 44. Angepasste Funktionen für Vergleichselemente (Forts.)

Funktionsname	Beschreibung
ANY_LE(arg1, arg2)	<p>Überprüft ein Wiederholungsattribut auf alle Werte, die kleiner-gleich dem angegebenen Wert sind. Zwei Argumente sind erforderlich:</p> <p><b>arg1</b> Gibt den Namen einer Spalte an, die ein Wiederholungsattribut darstellt.</p> <p><b>arg2</b> Gibt den zu vergleichenden Wert an.</p> <p>Beispiel:  <code>... WHERE DCTM.ANY_LE(num_approvers,2)=1</code></p>
ANY_GE(arg1, arg2)	<p>Überprüft ein Wiederholungsattribut auf alle Werte, die größer-gleich dem angegebenen Wert sind. Zwei Argumente sind erforderlich:</p> <p><b>arg1</b> Gibt den Namen einer Spalte an, die ein Wiederholungsattribut darstellt.</p> <p><b>arg2</b> Gibt den zu vergleichenden Wert an.</p> <p>Beispiel:  <code>... WHERE DCTM.ANY_GE(num_approvers,1)=1</code></p>
ANY_IN(arg1, arg2 – arg11)	<p>Überprüft ein Wiederholungsattribut auf zehn Werte in einer angegebenen Liste mit Werten. Drei bis elf Argumente desselben Datentyps können angegeben werden:</p> <p><b>arg1</b> Gibt den Namen einer Spalte an, die ein Wiederholungsattribut darstellt.</p> <p><b>arg2–arg11</b> Gibt eine Liste der zu vergleichenden Werte an, die durch Komma voneinander getrennt sind.</p> <p>Beispiel:  <code>... WHERE DCTM.ANY_IN(authors,'Crick','Watson')=1</code></p> <p>Die maximal mögliche Anzahl von Werten in einer angepassten Funktion ANY_IN für Wiederholungsattribute ist 10 bei einer einzelnen Anweisung. Mehrere Anweisungen können mit OR verknüpft werden.</p>
ANY_LIKE(arg1, arg2)	<p>Überprüft ein Wiederholungsattribut auf alle Werte, die ähnlich dem angegebenen Wert sind. Zwei Argumente sind erforderlich:</p> <p><b>arg1</b> Gibt den Namen einer Spalte an, die ein Wiederholungsattribut darstellt.</p> <p><b>arg2</b> Gibt das zu vergleichende Muster an; untergeordnete Zeichenfolgen werden in einfache Anführungszeichen eingeschlossen.</p> <p>Beispiel:  <code>... WHERE DCTM.ANY_LIKE(authors,'Dave Win%')=1</code>  <code>OR DCTM.ANY_LIKE(keywords,'%\_%')=1</code></p> <p>Die Escape-Klausel wird in ANY_LIKE()-Vergleichselementen nicht unterstützt.</p>

Tabelle 44. Angepasste Funktionen für Vergleichselemente (Forts.)

Funktionsname	Beschreibung
ANY_NOT_LIKE(arg1, arg2)	<p>Überprüft ein Wiederholungsattribut auf alle Werte, die nicht ähnlich dem angegebenen Wert sind. Zwei Argumente sind erforderlich:</p> <p><b>arg1</b> Gibt den Namen einer Spalte an, die ein Wiederholungsattribut darstellt.</p> <p><b>arg2</b> Gibt das zu vergleichende Muster an; untergeordnete Zeichenfolgen werden in einfache Anführungszeichen eingeschlossen.</p> <p>Beispiel:  <pre>... WHERE DCTM.ANY_NOT_LIKE(authors, 'Dave Win%')=1 OR DCTM.ANY_NOT_LIKE(keywords, '%\_%')</pre> </p> <p>Die Escape-Klausel wird in ANY_NOT_LIKE()-Vergleichselementen nicht unterstützt.</p>
ANY_NULL(arg)	<p>Überprüft ein Wiederholungsattribut auf den Wert IS NULL. Ein erforderliches Argument muss angegeben werden, das den Namen des Wiederholungsattributs oder des nur einen Wert enthaltenden Attributs DATE bzw. TIMESTAMP angibt.</p> <p>Beispiel:  <pre>... WHERE DCTM.ANY_NULL(authors)=1</pre> </p>
ANY_NOT_NULL(arg)	<p>Überprüft ein Wiederholungsattribut auf den Wert IS NOT NULL. Ein erforderliches Argument muss angegeben werden, das den Namen des Wiederholungsattributs darstellt.</p> <p>Beispiel:  <pre>... WHERE DCTM.ANY_NOT_NULL(authors)=1</pre> </p>
ANY_SAME_INDEX(arg1 – arg10)	<p>Überprüft Wiederholungsattribute auf Werte mit demselben Index für jedes Attribut. Zwei bis zehn der anderen ANY_xx()-Funktionen können angegeben werden.</p> <p>Im folgenden Beispiel wird überprüft, ob ein Dokument mindestens einen Autor mit dem Namen Ken aufweist, der nicht zu UCD gehört.</p> <pre>... WHERE DCTM.ANY_SAME_INDEX( ANY_EQ(author_name, 'Ken'), DCTM.ANY_NE(author_affiliation, 'UCD'))=1</pre> <p>Die maximal mögliche Anzahl von Überprüfungen auf Werte derselben Indexstufe für Wiederholungsattribute ist 10. Bei den Überprüfungen muss es sich um AND-Überprüfungen handeln, die von links nach rechts bewertet werden.</p>

Tabelle 44. Angepasste Funktionen für Vergleichselemente (Forts.)

Funktionsname	Beschreibung
CABINET(arg) und CABINET_TREE(arg)	<p>Ein erforderliches Argument muss angegeben werden, das den vollständig qualifizierten Namen einer Docbase-Ablage darstellt.</p> <p>Beispiel:</p> <pre>... WHERE DCTM.CABINET('/Tools')=1 ... WHERE DCTM.CABINET_TREE('/MyDocs')=1</pre> <p>Verwenden Sie mehrere Exemplare von CABINET und CABINET_TREE, um mehrere Ablagen anzugeben.</p> <p>Beispiel:</p> <pre>... WHERE DCTM.CABINET('/Tools')=1 OR DCTM.CABINET_TREE('/Parts')=1</pre>
FOLDER(arg) und FOLDER_TREE(arg)	<p>Ein erforderliches Argument muss angegeben werden, das den vollständig qualifizierten Namen eines Docbase-Ordners bzw. einer Docbase-Ablage darstellt.</p> <p>Beispiel:</p> <pre>... DCTM.FOLDER('/Tools/Drills')=1 ... DCTM.FOLDER_TREE('/MyDocs/WhitePapers')=1</pre> <p>Verwenden Sie mehrere Exemplare von FOLDER und FOLDER_TREE, um mehrere Ordner anzugeben.</p> <p>Beispiel:</p> <pre>... DCTM.FOLDER('/Tools/Drills')=1 OR DCTM.FOLDER_TREE('/Animals/Horses')=1</pre>
RENDITION_FORMAT (format)	<p>Verwendet die Pseudospalten GET_RENDITION und GET_RENDITION_DEL, um das Format des abzurufenden Ausgabeformats zu ermitteln. Zur Angabe des Formats ist ein Zeichenfolgeargument erforderlich.</p> <p>Das folgende Beispiel ruft ein Dokument im PDF-Format ab:</p> <pre>SELECT get_rendition FROM .... WHERE DCTM.RENDITION_FORMAT('pdf')=1</pre>
USER(1)	<p>Vergleicht einen Wert mit der Documentum-Autoren-ID des aktuellen Benutzers. Aufgrund einer Einschränkung bei DB2 wird die angepasste Funktion USER mit einem ganzzahligen Argument definiert, das nicht verwendet wird.</p> <p>Beispiel:</p> <pre>... WHERE approver = DCTM.USER(1)</pre> <p>Verwenden Sie die Anweisung CREATE USER MAPPING, um die Documentum-Autoren-ID entsprechend der DB2-Autoren-ID zu definieren.</p>



Tabelle 44. Angepasste Funktionen für Vergleichselemente (Forts.)

Funktionsname	Beschreibung
SEARCH_WORDS(arg)	<p>Ein erforderliches Zeichenfolgeargument muss angegeben werden, bei dem es sich um eine Liste einzelner Wörter handelt. Diese Wörter müssen in einfache Anführungszeichen eingeschlossen und durch AND, OR oder NOT verbunden werden, die Vorrangstellung muss durch runde Klammern gesteuert werden. Die Wörter dürfen keine Leerzeichen enthalten und müssen in einfache Anführungszeichen eingeschlossen sein.</p> <p>Beispiel:</p> <pre>... DCTM.SEARCH_WORDS(''yeast'' AND (''bread'' OR ''cake'') AND NOT ''wedding'' )=1</pre>
SEARCH_TOPIC(arg)	<p>Ein erforderliches Zeichenfolgeargument muss angegeben werden, bei dem es sich um eine Verity-TOPIC-Abfrageanweisung handelt, die wörtlich an Documentum und Vertiy übergeben werden muss.</p> <p>Beispiel:</p> <pre>... WHERE DCTM.SEARCH_TOPIC(''quick'')=1</pre>

Dies war die letzte Task in dieser Tasksequenz.

**Zugehörige Referenzen:**

- „CREATE FUNCTION (Sourced or Template) statement“ in *SQL Reference, Volume 2*

## Documentum-Datenquelle - Beispielabfragen

Nach der Registrierung des Wrappers können Sie SQL-Abfragen für die Documentum-Datenquelle ausführen. Dieser Abschnitt enthält eine Reihe von Beispielabfragen.

Um Abfragen auszuführen, verwenden Sie den Kurznamen und die definierten Kurznamenspalten in Ihren SQL-Anweisungen in derselben Weise wie reguläre Tabellennamen und Tabellenspalten.

**Einschränkungen für Verbindungen:**

Für jede Verbindung, die eine DB2-Anwendung zu einer DB2-Datenbank herstellt, kann der Documentum-Wrapper maximal 10 gleichzeitige Documentum-Sitzungen unterstützen; jede dieser Sitzungen kann ihrerseits bis zu 10 Documentum-Abfragen gleichzeitig bearbeiten.

Eine einzelne DB2-Anwendung kann mehrere Abfragen gleichzeitig ausführen. Die Lebensdauer einer Abfrage beginnt mit ihrer Übergabe an DB2 und endet, wenn der entsprechende Cursor auf der Ergebnismenge geschlossen wird.

Zu jedem Zeitpunkt kann von einem Documentum-Server für die gesamte zu diesem Zeitpunkt aktive Gruppe von Abfragen auf höchstens 10 Kurznamen verwiesen werden. Kurznamen, die in mehreren Abfragen erwähnt werden oder auf die in einer einzelnen Abfrage mehrfach verwiesen wird, müssen für jedes Auftreten einmal gezählt werden.

## Das Vergleichselement LIKE:

Der Documentum-Server und DB2 verarbeiten das Vergleichselement LIKE unterschiedlich. Wird ein Vergleichselement LIKE an den Documentum-Server ausgelagert, so findet die Documentum-Semantik Anwendung. Enthält im folgenden Beispiel die Spalte c1 eine Zeichenfolge der Länge null, so ist das Vergleichselement für Documentum wahr und für DB2 falsch.

```
c1 LIKE '%'
```

## Beispielabfragen:

Mit der folgenden Abfrage werden alle Docbase-Dokumente für den Dokumentnamen 'Test Document' angezeigt:

```
SELECT object_name
FROM std_doc
WHERE object_name='Test Document';
```

In der folgenden Abfrage wird die angepasste Funktion ANY\_EQ verwendet, um alle Dokumente anzuzeigen, bei denen einer der Autoren 'Joe Doe' heißt.

```
SELECT object_name
FROM std_doc
WHERE DCTM.ANY_EQ(author, 'Joe Doe')=1
```

In der folgenden Abfrage werden die Funktionen FOLDER\_TREE und SEARCH\_WORDS verwendet, um alle Dokumente in der Ablage 'Approved' (Genehmigt) zu suchen, die den Text "protein" enthalten.

```
SELECT object_name
FROM std_doc
WHERE DCTM.FOLDER_TREE('/Approved')=1
      AND DCTM.SEARCH_WORDS('protein')=1
```

In der folgenden Abfrage werden die Pseudospalte GET\_FILE und die angepassten Funktionen FOLDER\_TREE und ANY\_IN verwendet, um die Namen der Dateien auf dem DB2-Server abzurufen, in denen der Inhalt für alle Dokumente, die einen der angeführten Autoren enthalten, in der Ablage 'Approved' ('genehmigt') gespeichert wurde.

```
SELECT object_name, object_id, get_file
FROM std_doc
WHERE DCTM.FOLDER_TREE('/Approved')=1
      AND DCTM.ANY_IN(author, 'Mary Black', 'Joe Carson', 'Peter Miller')=1
```

## Zugehörige Tasks:

- „Registrieren der angepassten Funktionen für den Documentum-Wrapper“ auf Seite 206

## Zugehörige Referenzen:

- „Doppelte Definition von Wiederholungsattributen (Documentum-Wrapper)“ auf Seite 205
- „Zugriffssteuerung für den Documentum-Wrapper“ auf Seite 214

## Zugriffssteuerung für den Documentum-Wrapper

Abfragen sind abhängig von den Berechtigungen des Benutzers innerhalb der Database. Nur die Dokumente, für die der Benutzer mindestens über Lesezugriff verfügt, sind in den entsprechenden Abfrageergebnissen enthalten.

### Zugehörige Referenzen:

- „Modell der Dateizugriffssteuerung für den Wrapper für Dateien mit Tabellenstruktur“ auf Seite 410
- „Modell für die Dateizugriffssteuerung für den Excel-Wrapper“ auf Seite 256

---

## Nachrichten für den Documentum-Wrapper

In diesem Abschnitt sind Nachrichten und die zugehörigen Beschreibungen aufgeführt, die bei der Verwendung des Wrappers für Documentum angezeigt werden können.

*Tabelle 45. Vom Documentum-Wrapper ausgegebene Nachrichten*

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "sqlno_crule_save_plans [100]:rc (-2144272209) Leere Planliste festgestellt".)	Die an DB2 übergebene SQL-Abfrage konnte vom Wrapper nicht verarbeitet werden. Korrigieren Sie die Syntax und übergeben Sie die Abfrage erneut.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "dmAPI-Exec fehlgeschlagen: [DM_QUERY_E_BAD_QUAL] Fehler: "Die Attributkennung A0 für das Attribut <spaltenname> ist keine gültige Kennung".)	Ein nicht korrekter Documentum-Typ oder eine nicht korrekte registrierte Tabelle wurde für die Kurznamenoption REMOTE_OBJECT eingegeben. Ändern Sie den Kurznamen, so dass der korrekte Documentum-Objekttyp bzw. die korrekte registrierte Tabelle verwendet wird.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Ungültige NULL-Spalte angegeben".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.

Tabelle 45. Vom Documentum-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Die Spezifikation für den Kurznamen ist leer".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Das Ausgabeobjekt ist leer oder unvollständig".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Unerwartete Anzahl von Spalten angefordert".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Keine Spalteninformationen gefunden".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Nicht unterstützter Spaltentyp angefordert".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Spaltendefinition nicht korrekt".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.

Tabelle 45. Vom Documentum-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Inkonsistenter Typ; DB2-Anforderung != Kurznamentyp".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Ausgabeparameter ist nicht NULL".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Abfrageausgabvariable ist nicht NULL".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Ungültige Länge der Zeitmarke".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Inkonsistente Spaltenanzahl".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Zugriff auf Daten beim Konvertieren von Werten nicht möglich".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.

Tabelle 45. Vom Documentum-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "DMCL-Client konnte nicht initialisiert werden".)	Der Documentum-Client kann nicht initialisiert werden. Benachrichtigen Sie den Systemadministrator.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Get_User gab den Wert NULL zurück".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Get_Local_User gab den Wert NULL zurück".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Transaktionsbeginn (begintrans) fehlgeschlagen".)	Documentum berichtete, dass 'begintrans' fehlgeschlagen ist. Benachrichtigen Sie den Systemadministrator.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Eingabeparameter war nicht NULL".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Dctm-Funktionen müssen wie DCTM.function(...) =1 sein.")	Sie haben als RHS des Vergleichselements für eine Dctm-Funktion nicht =1 verwendet. Korrigieren Sie die Syntax und führen Sie die Abfrage erneut aus.

Tabelle 45. Vom Documentum-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Ungültige Spaltennummer angefordert".)	Interner Programmierfehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1881N	"DELIMITER" ist keine gültige Option "COLUMN" für "<spaltenname>"	Die Option DELIMITER wurde für Spalte <spaltenname> angegeben. Die Option IS_REPEATING wurde jedoch nicht angegeben.
SQL1882N	Die Option "RDBMS_TYPE" der Art "SERVER" kann nicht auf "<optionswert>" für "<servername>" eingestellt werden.	Der für die Serveroption RDBMS_TYPE angegebene Wert ist ungültig. Es muss einer der folgenden Werte angegeben werden: DB2, INFORMIX, ORACLE, SQLSERVER oder SYBASE.
SQL1882N	Die Option "TRANSACTIONS" der Art "SERVER" kann nicht auf "<optionswert>" für "<servername>" eingestellt werden.	Der für die Serveroption TRANSACTIONS angegebene Wert ist ungültig. Es muss einer der folgenden Werte angegeben werden: NONE, QUERY, PASSTHRU oder ALL.
SQL1882N	Die Option "IS_REG_TABLE" der Art "NICKNAME" kann nicht auf "<optionswert>" für "<kurzname>" eingestellt werden.	Der für die Kurznamenoption IS_REG_TABLE angegebene Wert ist ungültig. Es muss einer der folgenden Werte angegeben werden: 'Y' oder 'N'.
SQL1882N	Die Option "ALL_VERSIONS" der Art "NICKNAME" kann nicht auf "<optionswert>" für "<kurzname>" eingestellt werden.	Der für die Kurznamenoption ALL_VERSIONS angegebene Wert ist ungültig. Es muss einer der folgenden Werte angegeben werden: 'Y' oder 'N'.
SQL1882N	Die Option "OS_TYPE" der Art "SERVER" kann nicht auf "<optionswert>" für "<servername>" eingestellt werden.	Der für die Serveroption OS_TYPE angegebene Wert ist ungültig. Es muss einer der folgenden Werte angegeben werden: AIX, HPUX, SOLARIS oder WINDOWS.
SQL1882N	Die Option "FOLDERS" der Art "NICKNAME" kann nicht auf "<optionswert>" für "<kurzname>" eingestellt werden.	Der für die Kurznamenoption FOLDERS angegebene Wert ist ungültig. Er kann nicht für Tabellen angegeben werden, bei denen für IS_REG_TABLE der Wert 'Y' angegeben ist.
SQL1882N	Die Option "VERSIONS" der Art "NICKNAME" kann nicht auf "<optionswert>" für "<kurzname>" eingestellt werden.	Der für die Kurznamenoption VERSIONS angegebene Wert ist ungültig. Es muss einer der folgenden Werte angegeben werden: 'Y' oder 'N'. Die Option VERSIONS kann für eine Tabelle nicht auf 'Y' gesetzt werden, bei der die Option IS_REG_TABLE auf 'Y' gesetzt ist.

Tabelle 45. Vom Documentum-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Ungültiger Spaltenname, IS_REG_TABLE oder IS_REPEATING im Kurznamen angegeben"	Überprüfen Sie, ob in der Kurznamenangabe die Optionen IS_REG_TABLE, IS_REPEATING und REMOTE_NAME sowie die Spaltennamen korrekt angegeben sind.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Die Protokolldatei konnte nicht für den Debug-Vorgang geöffnet werden"	Auf die für die Fehlerbehebung verwendete Datei kann nicht zugegriffen werden. Benachrichtigen Sie den Systemadministrator.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Es darf nur eine Suchbedingung angegeben werden."	Pro Abfrage kann nur eine angepasste Suchfunktion angegeben werden.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode: "Das Verzeichnis für die Inhaltsdatei konnte nicht erstellt werden."	Stellen Sie sicher, dass der DB2-Agent Schreibzugriff auf das Zielverzeichnis hat.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode: "Die Berechtigungen für die Inhaltsdatei konnten nicht geändert werden."	Stellen Sie sicher, dass der DB2-Agent Schreibzugriff auf das Zielverzeichnis mit den Inhaltsdateien hat.
SQL5182N	Die erforderliche Umgebungsvariable "DMCL_CONFIG" wurde nicht definiert.	Weder DOCUMENTUM noch DMCL_CONFIG wurde als Umgebungsvariable festgelegt. Definieren Sie diese in der Datei db2dj.ini.

**Zugehörige Konzepte:**

- „Nachrichten - Einführung“ in *Fehlernachrichten Band 1*

**Zugehörige Referenzen:**

- „SQLSTATE-Nachrichten“ in *Fehlernachrichten Band 2*





---

## Kapitel 11. Konfigurieren des Zugriffs auf Entrez-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Entrez-Datenquellen gespeichert sind. Der Zugriff auf Entrez-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält folgende Informationen:

- Erläuterungen zu Entrez.
- Liste der auszuführenden Tasks.
- Beispiele zu den benötigten SQL-Anweisungen.
- Liste der dem Entrez-Wrapper zugeordneten Fehlermeldungen.

---

### Was ist Entrez?

Entrez ist ein Abfrage- und Abrufsystem, das vom NCBI (National Center for Biotechnology Information) entwickelt wurde. Mit Hilfe von Entrez können Sie auf mehrere miteinander verbundene Datenbanken zugreifen, die vom NCBI verwaltet werden.

Zu diesen Datenbanken gehören:

- PubMed (biomedizinische Literatur)
- Nucleotide (eine Sequenzdatenbank, auch 'GenBank' genannt)
- OMIM ('Online Mendelian Inheritance in Man', Onlinedatenbank zur Mendelschen Vererbungstheorie der John Hopkins-Universität)
- Genome (vollständige Genom-Baugruppen)

Auf alle diese Entrez-Datenbanken kann über eine einheitliche Gruppe webbasierter Tools zugegriffen werden. Der Entrez-Wrapper verwendet diese Tools zum Eingliedern der Entrez-Datenbanken in die DB2<sup>®</sup>-Umgebung. Die Entrez-Schnittstelle unterstützt viele verschiedene Datenbanken, wohingegen der Entrez-Wrapper nur die Datenbanken PubMed und Nucleotide unterstützt.

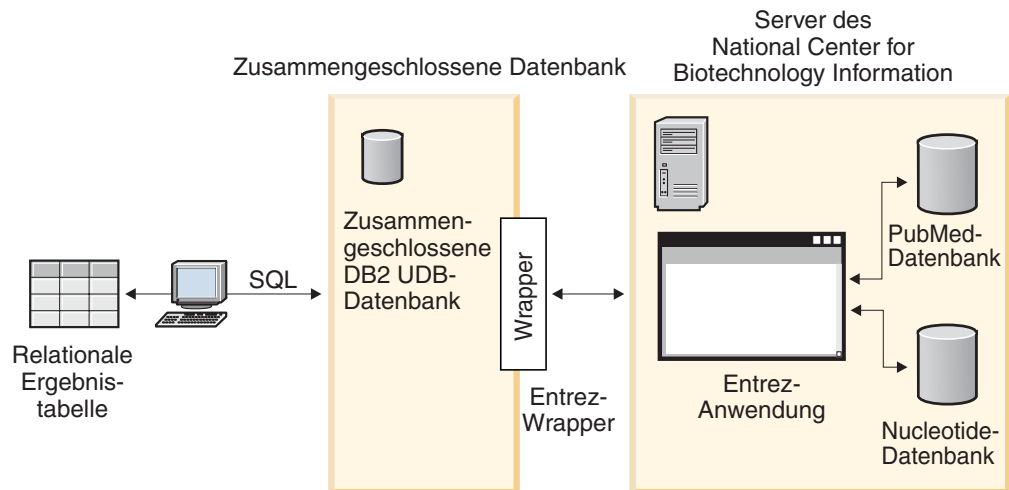


Abbildung 22. Funktionsweise des Entrez-Wrappers

Viele Elemente des Entrez-Wrappers sind bei allen Datenbanken gleich. Zu diesen Elementen gehören:

- Verbindung mit NCBI über das Web und die Entrez-Dienstprogramme ESearch und EFetch
- Zuordnung hierarchischer XML-Daten in relationale Tabellen
- Verknüpfungen zwischen Tabellen über die XML-Wrappertechnologie

**Zugehörige Tasks:**

- „Hinzufügen von Entrez-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 222

---

## Hinzufügen von Entrez zu einem Server mit zusammengeschlossenen Datenbanken

### Hinzufügen von Entrez-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Entrez-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

| Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken  
 | auf Entrez-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-  
 | Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der  
 | Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusam-  
 | mengeschlossenen Datenbanken führt.

Der Entrez-Wrapper bietet Zugriff auf PubMed- und Nucleotide-Datenquellen in Netzwerken, die Firewalls mit Proxys verwenden. Unterstützte Proxys sind: HTTP, SOCKS4 und SOCKS5.

**Voraussetzungen:**

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.

- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.

#### Vorgehensweise:

Gehen Sie wie folgt vor, um Entrez-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Registrieren Sie angepasste Funktionen für den Entrez-Wrapper.
2. Registrieren Sie den Wrapper.
3. Registrieren Sie die Serverdefinition.
4. Registrieren Sie Kurznamen für Entrez-Datenbanken.

#### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

#### Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Registrieren von angepassten Funktionen für den Entrez-Wrapper“ auf Seite 223

#### Zugehörige Referenzen:

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## Registrieren von angepassten Funktionen für den Entrez-Wrapper

Die Registrierung von angepassten Funktionen für den Entrez-Wrapper ist Teil der umfangreicheren Task des Hinzufügens von Entrez zu einem System zusammengeschlossener Datenbanken. Nachdem Sie die angepassten Funktionen registriert haben, muss der Wrapper registriert werden.

#### Einschränkungen:

- Alle angepassten Funktionen für den Entrez-Wrapper müssen mit dem Schemanamen `entrez` registriert werden.
- Jede angepasste Funktion muss einmal für jede DB2-Datenbank registriert werden, in der der Entrez-Wrapper installiert ist.

#### Vorgehensweise:

Um angepasste Funktionen zu registrieren, geben Sie die Anweisung `CREATE FUNCTION` mit den Schlüsselwörtern `AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION` aus.

Der vollständig qualifizierte Name jeder Funktion ist `entrez.function_name`.

Im folgenden Beispiel wird eine Version der Funktion `CONTAINS` registriert:

```
CREATE FUNCTION entrez.contains (varchar(), varchar())
  RETURNS INTEGER AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION;
```

Um angepasste Funktionen zu registrieren, verwenden Sie die Beispieldatei `create_function_mappings.ddl`. Die Beispieldatei befindet sich in dem Pfad, in dem DB2 Information Integrator installiert ist, im Verzeichnis `samples/lifesci/entrez`. Die Beispieldatei enthält Definitionen für die einzelnen angepassten Funktionen. Sie können diese DDL-Datei zum Registrieren der angepassten Funktionen für jede zusammengeschlossene Datenbank ausführen, für die der Entrez-Wrapper installiert ist.

Die nächste Task in dieser Tasksequenz ist das Registrieren des Entrez-Wrappers.

**Zugehörige Referenzen:**

- „CREATE FUNCTION (Sourced or Template) statement“ in *SQL Reference, Volume 2*
- „Angepasste Funktionen und Entrez-Abfragen“ auf Seite 230
- „Angepasste Funktionstabelle - Entrez-Wrapper“ auf Seite 224

## Angepasste Funktionstabelle - Entrez-Wrapper

Verwenden Sie die Anweisung `CREATE FUNCTION`, um die angepassten Entrez-Funktionen zu registrieren.

Die folgende Tabelle listet die angepassten Entrez-Funktionen und die Datentypen der Argumente auf, die Sie beim Registrieren der Funktionen angeben. Das erste in der Funktion angegebene Argument wird für den Spaltennamen einer Spalte mit Kennung bereitgestellt. Das zweite in der Funktion angegebene Argument ist der Suchbegriff.

*Tabelle 46. Angepasste Funktionen für den Entrez-Wrapper*

<b>Funktion</b>	<b>Beschreibung</b>
<code>entrez.contains (varchar(), varchar())</code>	Durchsucht eine Spalte mit Kennung unter Verwendung des angegebenen Begriffs.
<code>entrez.contains (integer, varchar())</code>	
<code>entrez.contains (smallint, varchar())</code>	
<code>entrez.contains (real, varchar())</code>	
<code>entrez.contains (double, varchar())</code>	
<code>entrez.contains (date, varchar())</code>	
<code>entrez.contains (time, varchar())</code>	
<code>entrez.contains (char(), varchar())</code>	
<code>entrez.contains (timestamp(), varchar())</code>	
<code>entrez.search_term (char(), varchar())</code>	Übergibt einen Entrez-Suchbegriff direkt an die Entrez- Suchsteuerkomponente.

Um angepasste Funktionen zu registrieren, verwenden Sie die Beispieldatei `create_function_mappings.ddl`. Diese Beispieldatei ist im Verzeichnis `samples/lifesci/entrez` installiert.

**Zugehörige Tasks:**

- „Registrieren von angepassten Funktionen für den Entrez-Wrapper“ auf Seite 223

**Zugehörige Referenzen:**

- „Angepasste Funktionen und Entrez-Abfragen“ auf Seite 230

## Registrieren des Entrez-Wrappers

Die Registrierung des Entrez-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von Entrez-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf Entrez-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Bibliotheksdatei des Wrappers aus.

Um beispielsweise einen Wrapper mit dem Namen `entrez_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER entrez_wrapper LIBRARY 'libdb2lsentrez.a'  
  OPTIONS(EMAIL 'jeff@someplace.com');
```

Zum Registrieren eines Entrez-Wrappers müssen Sie eine E-Mail-Adresse angeben. Diese E-Mail-Adresse wird in alle Abfragen eingefügt und ermöglicht es NCBI, bei auftretenden Problemen mit Ihnen Kontakt aufzunehmen, z. B. bei Überlastung der NCBI- Server durch zu viele Abfragen.

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung CREATE WRAPPER finden Sie in der Liste der Bibliotheksdateien von Entrez-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinition für Entrez-Datenquellen.

### Zugehörige Referenzen:

- „Bibliotheksdateien des Entrez-Wrappers“ auf Seite 225
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien des Entrez-Wrappers

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den Entrez-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, werden 3 Bibliotheksdateien zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden folgende Wrapperbibliotheksdateien zum Verzeichnispfad hinzugefügt: `libdb2lsentrez.a`, `libdb2lsentrezF.a` und `libdb2lsentrezU.a`.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

Tabelle 47. Positionen und Dateinamen von Entrez-Wrapperbibliotheken

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2lsentrez.a
Linux	/opt/IBM/db2/V8.1/lib	libdb2lsentrez.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lsentrez.so
Windows	%DB2PATH%\bin	db2lsentrez.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist C:\Program Files\IBM\SQLLIB.

#### Zugehörige Tasks:

- „Registrieren des Entrez-Wrappers“ auf Seite 225

## Registrieren des Servers für eine Entrez-Datenquelle

Die Registrierung des Servers für eine Entrez-Datenquelle ist Teil der umfangreicheren Task des Hinzufügens von Entrez zu einem Server mit zusammengeschlossenen Datenbanken.

Nach der Registrierung des Wrappers muss eine entsprechende Serverdefinition registriert werden.

Die durch eine bestimmte Datenquelle dargestellte Datenbank ('PubMed' oder 'Nucleotide') wird durch den in der Anweisung CREATE SERVER angegebenen Wert für den Servertyp identifiziert. Dieser Wert des Servertyps steuert die Struktur sämtlicher erstellter Kurznamen.

#### Vorgehensweise:

Um den Entrez-Server für das System zusammengeschlossener Datenbanken zu registrieren, geben Sie die Anweisung CREATE SERVER aus.

Um beispielsweise einen Server mit dem Namen `pubmed_server1` für den Wrapper `entrez_wrapper` zu registrieren, geben Sie die folgende Anweisung aus:

```
CREATE SERVER pubmed_server1
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper;
```

Um zusätzlich einen Server mit dem Namen `nucleotid_server1` für den Wrapper `entrez_wrapper` zu registrieren, geben Sie die folgende Anweisung aus:

```
CREATE SERVER nucleotid_server1
  TYPE NUCLEOTIDE
  VERSION 1.0
  WRAPPER entrez_wrapper;
```

## Beschränken der Anzahl der für Entrez-Abfragen abgerufenen Zeilen

Mit Hilfe der Serveroption `MAX_ROWS` kann die Anzahl der Zeilen begrenzt werden, die für eine Abfrage zurückgegeben werden, die den Entrez-Wrapper verwendet.

Im Gegensatz zur Klausel `FETCH FIRST N ROWS ONLY` einer SQL-Anweisung, die die Anzahl der Zeilen begrenzt, die an einen Benutzer oder eine Anwendung zurückgegeben werden, lässt sich mit Hilfe der Serveroption `MAX_ROWS` die Anzahl der Zeilen begrenzen, die von der NCBI-Website abgerufen werden können.

Der Wert der Option `MAX_ROWS` wird stets als Obergrenze (Maximum) für die Anzahl der Zeilen verwendet, die eine Abfrage abrufen kann. Versucht eine Abfrage, mehr Zeilen abzurufen, als in der Option `MAX_ROWS` angegeben, wird die Ergebnismenge abgeschnitten und eine Warnung ausgegeben.

Sie können die Serveroption `MAX_ROWS` beim Erstellen eines Servers festlegen oder den Optionswert mit Hilfe der Anweisung `ALTER SERVER` ändern.

Die Serveroption `MAX_ROWS` ist nicht erforderlich. Wird die Option nicht festgelegt, wird der Standardwert verwendet. Der verwendete Standardwert hängt vom jeweiligen Betriebssystem ab. Bei Microsoft Windows-Betriebssystemen lautet der Standardwert 2000 Zeilen. Bei UNIX-Betriebssystemen lautet der Standardwert 5000 Zeilen.

Als Wert können lediglich positive Zahlen und 0 (Null) angegeben werden. Wird die Option auf 0 (Null) gesetzt, können Abfragen eine unbegrenzte Anzahl an Zeilen von der NCBI-Website abrufen. Wird für die Serveroption `MAX_ROWS` jedoch 0 (Null) oder eine sehr große Zahl angegeben, kann sich dies möglicherweise auf die Abfrageleistung auswirken.

## Zugreifen auf Entrez über einen Proxy-Server

Um auf Entrez-Datenquellen über einen Proxy-Server zuzugreifen, müssen Sie beim Erstellen der Serverdefinition Optionen angeben. Diese von Ihnen angegebenen Optionen sind abhängig von dem Typ des Proxy-Servers, auf den Sie zugreifen möchten.

### Beispiel für das Registrieren einer Serverdefinition für einen HTTP-Proxy-Server:

Verwenden Sie folgende Anweisung, um eine Serverdefinition zu registrieren und einen HTTP-Proxy-Server anzugeben:

```
CREATE SERVER pubmed_server_h
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper
  OPTIONS (PROXY_TYPE 'HTTP', PROXY_SERVER_NAME 'proxy_h',
          PROXY_SERVER_PORT '8080');
```

### Beispiel für das Registrieren einer Serverdefinition für einen SOCKS4-Proxy-Server:

Verwenden Sie folgende Anweisung, um eine Serverdefinition zu registrieren und einen SOCKS4-Proxy-Server anzugeben:

```
CREATE SERVER pubmed_server_s4
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS4', PROXY_SERVER_NAME 'proxy_4',
          PROXY_SERVER_PORT '1080');
```



### Beispiel für das Registrieren einer Serverdefinition für einen SOCKS5-Proxy-Server ohne Authentifizierungsinformationen:

Verwenden Sie folgende Anweisung, um eine Serverdefinition zu registrieren und einen SOCKS5-Proxy-Server ohne Authentifizierungsinformationen anzugeben:

```
CREATE SERVER pubmed_server_s5
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS5', PROXY_SERVER_NAME 'proxy_5',
          PROXY_SERVER_PORT '1081');
```

### Beispiel für das Registrieren einer Serverdefinition für einen SOCKS5-Proxy-Server mit Authentifizierungsinformationen:

Verwenden Sie folgende Anweisung, um eine Serverdefinition zu registrieren und einen SOCKS5-Proxy-Server mit Authentifizierungsinformationen anzugeben:

```
CREATE SERVER pubmed_server_s5a
  TYPE PUBMED
  VERSION 1.0
  WRAPPER entrez_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS5', PROXY_SERVER_NAME 'proxy_5',
          PROXY_SERVER_PORT '1081', PROXY_AUTHID 'Khalid',
          PROXY_PASSWORD 'aaa', );
```

Die nächste Task in dieser Tasksequenz ist das Registrieren von Kurznamen für Entrez-Datenquellen.

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für Entrez-Datenquellen“ auf Seite 228

#### Zugehörige Referenzen:

- „CREATE SERVER-Anweisungsargumente - Entrez-Wrapper“ auf Seite 617

## Registrieren von Kurznamen für Entrez-Datenquellen

Das Registrieren der Kurznamen für Entrez-Datenquellen ist Teil der umfangreicheren Task des Hinzufügens von Entrez zu einem System zusammengesetzter Datenbanken.

#### Einschränkungen:

Das Schema einer Entrez-Datenbank wird durch den Wrapper festgelegt und kann nicht verändert oder erweitert werden. Für jede Datenbank gibt es eine festgelegte Menge von Tabellen mit einer festgelegten Liste von Spalten für jede Tabelle. Die Tabellen innerhalb einer Datenbank haben eine hierarchische Beziehung. Eine Tabelle ist allen anderen Tabellen innerhalb der Datenbank übergeordnet und wird als Root- oder Stammtabelle (übergeordnete Tabelle) bezeichnet. Alle anderen Tabellen in der Datenbank haben eine Elter-Kind-Beziehung, die auf die Stammtabelle zurückführt.

#### Vorgehensweise:

Geben Sie zum Registrieren von Kurznamen für Entrez-Datenquellen eine Anweisung CREATE NICKNAME aus.

Da die Liste der Spalten für die Kurznamen durch den Wrapper festgelegt wird, ist die grundlegende Syntax zum Erstellen von Nucleotide-Kurznamen recht einfach. Beispiel:

```
CREATE NICKNAME GBSeq FOR SERVER nuc1;
CREATE NICKNAME GBFeatures FOR SERVER nuc1;
CREATE NICKNAME GBIntervals FOR SERVER nuc1;
CREATE NICKNAME GBQualifiers FOR SERVER nuc1;
CREATE NICKNAME GBReference FOR SERVER nuc1;
```

Nachfolgend ist ein Beispiel für die grundlegende Syntax zum Erstellen von PubMed-Kurznamen aufgeführt:

```
CREATE NICKNAME pmarticles FOR SERVER pubmed_server;
CREATE NICKNAME PMACCESSION FOR SERVER pubmed_server;
CREATE NICKNAME PMCHEMICAL FOR SERVER pubmed_server;
CREATE NICKNAME PMMESH FOR SERVER pubmed_server;
CREATE NICKNAME PMCOMMENTS FOR SERVER pubmed_server;
CREATE NICKNAME PMARTICLEID FOR SERVER pubmed_server;
```

Der Kurzname ist gleichzeitig der Name der zu Grunde liegenden Tabelle.

Durch die Verwendung dieser Syntax werden die Kurznamen auf eine einzige Kurznamenfamilie pro DB2-Schema begrenzt. Sie können jedoch auch andere Namen mit Hilfe der Kurznamenoptionen REMOTE\_OBJECT und PARENT verwenden. Für einen Stammkurznamen ist nur die Option REMOTE\_OBJECT erforderlich. Für alle anderen Kurznamen müssen beide Optionen, REMOTE\_OBJECT und PARENT, zur Verfügung gestellt werden.

Das folgende Beispiel zeigt dieselbe Gruppe von Nucleotide-Kurznamen unter Verwendung der Umbenennungsfunktion:

```
CREATE NICKNAME NewSeq FOR SERVER nuc1 OPTIONS (REMOTE_OBJECT 'GBSEQ');
CREATE NICKNAME NewFeatures FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBFEATURES', PARENT 'NEWSEQ');
CREATE NICKNAME NewIntervals FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBINTERVALS', PARENT 'NEWFEATURES');
CREATE NICKNAME NewQualifiers FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBQUALIFIERS', PARENT 'NEWFEATURES');
CREATE NICKNAME NewReference FOR SERVER nuc1
  OPTIONS (REMOTE_OBJECT 'GBREFERENCE', PARENT 'NEWSEQ');
```

Dieses Beispiel zeigt dieselbe Gruppe von PubMed-Kurznamen unter Verwendung der Umbenennungsfunktion:

```
CREATE NICKNAME newpmarticles FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMARTICLES');
CREATE NICKNAME NEWPMACCESSION FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMACCESSION', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMCHEMICAL FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMCHEMICAL', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMMESH FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMMESH', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMCOMMENTS FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMCOMMENTS', PARENT 'NEWPMARTICLES');
CREATE NICKNAME NEWPMARTICLEID FOR SERVER pubmed_server
  OPTIONS (REMOTE_OBJECT 'PMARTICLEID', PARENT 'NEWPMARTICLES');
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der angepassten Funktionen für Entrez-Datenquellen.

#### **Zugehörige Tasks:**

- „Hinzufügen von Entrez-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 222

- „Registrieren von angepassten Funktionen für den Entrez-Wrapper“ auf Seite 223
- „Angaben von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75

**Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „PubMed-Schematabellen“ auf Seite 235
- „Nucleotide-Schematabellen“ auf Seite 240

---

## Abfragen und angepasste Funktionen für Entrez-Datenquellen

### Angepasste Funktionen und Entrez-Abfragen

Die Umgebung zusammenschlossener Datenbanken verwendet zwei Steuerkomponenten für die Abfrage. Für den Entrez-Wrapper sind diese Abfrage-Steuerkomponenten DB2 und Entrez. Abgesehen von einer einzigen Ausnahme geben Sie alle Vergleichselemente für die Entrez-Steuerkomponente über die angepassten Funktionen an. Für die DB2-Steuerkomponente werden alle Vergleichselemente über die relationalen Operatoren angegeben.

Die wichtigste angepasste Funktion ist ENTREZ.CONTAINS. Die Funktion CONTAINS erfordert ein Argument für die Suchbegriffsspalte und ein Argument für den Abfragetext. Das nachfolgende Beispiel zeigt eine Anweisung ENTREZ.CONTAINS statement:

```
ENTREZ.CONTAINS (suchbegriffsspalte, abfragetext)
```

Eine Kennung in der Spalte Q der Schematabellen identifiziert den Suchbegriff. Der Abfragetext muss in der geänderten Entrez-Abfragesyntax enthalten sein. Diese Syntax besteht aus Suchbegriffen, die durch Boolesche Operatoren (OR, AND und NOT) von einander getrennt und durch runde Klammern gruppiert sind. Die Syntax des CONTAINS-Abfragetextarguments unterscheidet sich dahingehend von der standardmäßigen Entrez-Abfragesyntax, dass Qualifikationsmerkmale für Suchbegriffe, wie z. B. [pd], nicht zulässig sind.

Die angepassten Funktionen werden in dem Entrez-Schema registriert, das zum Verweisen auf die Funktionen verwendet wird. Werden die angepassten Funktionen verwendet, muss deren Rückgabewert in einem Vergleichselement mit dem Wert 1 verglichen werden.

In einigen Fällen kann es vorkommen, dass DB2- und Entrez-Vergleichselemente so miteinander vermischt werden, dass sie nicht verarbeitet werden können. Dann wird die Fehlermeldung SQL0142N ("Nicht unterstützte SQL-Anweisung") generiert.

In der nachfolgenden Abfrage beispielsweise können die Teile des Vergleichselements, die vom Wrapper verarbeitet werden (die Aufrufe ENTREZ.CONTAINS) nicht von den Teilen getrennt werden, die von DB2 verarbeitet werden müssen (das relationale Vergleichselement für BaseCountA).

```
WHERE
ENTREZ.CONTAINS (Organism, 'drosophila') = 1
OR (BaseCountA > 10 AND ENTREZ.CONTAINS (Keywords, 'glop') = 1)
```

Für einige Suchfelder gibt es keine zugeordneten Spalten im Entrez-Schema. In der Nucleotide-Datenbank z. B. durchsucht der Term [ALL] alle durchsuchbaren Felder, während der Term [WORD] den gesamten, einem Eintrag zugeordneten freien Text durchsucht. Für diese Suchbegriffe werden Pseudospalten bereitgestellt. Wird in einer Auswahlliste auf eine Pseudospalte verwiesen, wird der Wert NULL zurückgegeben.

Durch Ausgeben der Masterfunktion ENTREZ.SEARCH\_TERM können Sie Abfragen ausführen, die ansonsten unter Umständen nicht möglich wären. Wenn Sie die Masterfunktion ENTREZ.SEARCH\_TERM angeben, muss dies die einzige angepasste Funktion der Abfrage sein. Für jede Abfrage kann nur eine Masterfunktion ENTREZ.SEARCH\_TERM pro Entrez-Kurzname angegeben werden. Ebenso können die Funktionen SEARCH\_TERM und CONTAINS innerhalb einer Abfrage nicht für denselben Kurznamen angegeben werden. Das erste Argument, die Spaltenangabe, muss die Primärschlüsselspalte für den übergeordneten Kurznamen sein. Das zweite Argument, der Abfragetext, ist ein Suchbegriff im Entrez-Format, der Qualifikationsmerkmale für Suchfelder enthält. Dieser Text wird unverändert an Entrez übergeben. Eine Ausnahme bilden hierbei URI-Escapeelemente, die gemäß der URI-Syntax geändert werden.

Das folgende Beispiel zeigt eine Abfrage mit einer WHERE-Klausel für einen PubMed-Kurznamen:

```
WHERE
ENTREZ.CONTAINS (authors, 'kaufmann OR ito AND NOT rakesh')
AND
(ENTREZ.CONTAINS (title, 'drosophila')
OR
ENTREZ.CONTAINS(alltext, 'drosophila OR "fruit fly"'))
```

In diesem Beispiel stellen authors, title und all text die individuellen Vergleichselemente dar.

Die individuellen Vergleichselemente werden dahingehend geändert, dass das Qualifikationsmerkmal hinter jedem Suchbegriff hinzugefügt wird. Dann werden die Begriffe mit runden Klammern gruppiert, um eine Vorrangstellung für Boolesche Operatoren für DB2 zu erzwingen. Auf Grund dieser Änderungen wird das Vergleichselement authors zu:

```
((kaufmann[auth] OR ito[auth]) AND (NOT (rakesh[auth])))
```

Das Vergleichselement title wird zu:

```
(drosophila[tit1])
```

Das Vergleichselement all text wird zu:

```
(drosophila[all] OR "fruit fly"[all])
```

Bei Kombination von individuellen Vergleichselementen wird die Vorrangstellung für Boolesche Operatoren für DB2 mit Hilfe von runden Klammern erhalten. Ausschließlich der Texttransformationen, die zum Ausdruck der Zeichenfolge als Teil einer URI erforderlich sind, lautet der letztlich an Entrez übergebene Suchbegriff wie folgt:

```
((kaufman[auth] OR ito[auth]) AND (NOT (rakesh[auth]))) AND
((drosophila[tit1]) OR (drosophila[all] OR "fruit fly"[all]))
```

#### **Zugehörige Referenzen:**

- „Angepasste Funktionstabelle - Entrez-Wrapper“ auf Seite 224

## Relationale Vergleichselemente für den Entrez-Wrapper

Der Entrez-Wrapper unterstützt relationale Vergleichselemente wie z. B. =, BETWEEN, LIKE und <> für Kurznamenspalten. Die Entrez-Suchsteuerkomponente verarbeitet jedoch nur einen geringen Anteil dieser relationalen Vergleichselemente. Nicht von der Entrez-Suchsteuerkomponente verarbeitete relationale Vergleichselemente werden von DB2<sup>®</sup> verarbeitet. Die Entrez-Suchsteuerkomponente verarbeitet Gleichheits- (=) und IN-Vergleichselemente für bestimmte ID-Spalten jedes Schemas. Mit Hilfe dieser Vergleichselemente kann der Entrez-Wrapper die Suchphase umgehen und direkt die Abrufphase ausführen. Beispiele für gültige Vergleichselemente sind:

```
WHERE pmid = '1234567'
```

```
WHERE medlineid IN ('1234567', '9191919')
```

In Vergleichselementen dieser Art verwendbare Spalten werden durch die Spalte F der Schematabellen identifiziert. Der Wert für diese Option muss Y lauten.

### Zugehörige Konzepte:

- „Ungültige WHERE-Klauseln für den Entrez-Wrapper“ auf Seite 232

### Zugehörige Tasks:

- „Entrez-Datenquelle - Beispielabfragen“ auf Seite 233
- „Registrieren von angepassten Funktionen für den Entrez-Wrapper“ auf Seite 223

## Ungültige WHERE-Klauseln für den Entrez-Wrapper

Der Entrez-Wrapper weist jede Abfrage zurück, die zu unqualifiziertem Durchsuchen der NCBI-Datenbank führen würde. Eine gültige WHERE-Klausel muss entweder ein Gleichheits- oder IN-Vergleichselement für die Primär-ID des Schemas enthalten, oder eine angepasste Funktion. Abfragen, die diese Kriterien nicht erfüllen, werden mit den Fehlercodes SQL0142N oder SQL30090N zurückgewiesen.

### Zugehörige Konzepte:

- „Relationale Vergleichselemente für den Entrez-Wrapper“ auf Seite 232

### Zugehörige Tasks:

- „Entrez-Datenquelle - Beispielabfragen“ auf Seite 233
- „Registrieren von angepassten Funktionen für den Entrez-Wrapper“ auf Seite 223

## Vereinfachung von Schemadatenelementen

Verschiedene Datenelemente werden bei der Darstellung durch das SQL-Schema in ein kanonisches Format umgewandelt. Diese Datenelemente enthalten Artikellisten, Namen und Datumsangaben.

### Artikellisten

Wenn nichts anderes vermerkt ist, werden die Artikel in Listen, deren Elemente in einer einzigen Spalte denormalisiert sind, durch ein Semikolon und ein einzelnes Leerzeichen voneinander getrennt. Wenn z. B. ein Eintrag die Schlüsselwörter 'dnaA gene', 'dnaN gene' und 'orf187' beinhaltet, enthält die entsprechende Schlüsselwortspalte den Wert 'dnaA gene; dnaN gene; orf187'.

## Namen

Namen in den NCBI-Schemas bestehen aus einem erforderlichen Nachnamen und einem von mehreren optionalen Elementen. Einige dieser optionalen Elemente können gemeinsam auftreten, andere schließen sich gegenseitig aus. Um einen Namen in kanonischem Format zu erstellen, ordnen Sie die Elemente in eine Rangfolge ein. In einer Rangfolge von hoch nach niedrig lauten diese Elemente:

- Vorname
- Vorname oder zweiter Vorname
- Initialen

Sie können Namen mit oder ohne Zuordnungen darstellen. Ohne Zuordnung lautet das Format eines Namens <nachname>, <vorname>, wobei <vorname> eines der optionalen Elemente darstellt. Wird das Element <vorname> nicht gefunden, entfällt das Komma. Eine Zuordnung kann mit dem Format (<zuordnung>) hinzugefügt werden.

Trennen Sie Namen in denormalisierten Listen durch ein Semikolon und ein Leerzeichen voneinander. Dies ist ein Beispiel für korrektes Trennen von Namen:

Parker, M. J.; Ranjan, K. A.

## Datumsangaben

Datumsangaben, vor allem Erscheinungsdaten, verfügen über eine Vielfalt an Formaten in den NCBI-Schemas. Zur Verwendung dieser Formate und zur Zulassung von Datumsvergleichen und Datumsarithmetik, wenn möglich, werden Datumsangaben im SQL-Schema in zwei Formaten dargestellt. Zum einen kann ein Datum als Zeichenfolge dargestellt werden. Zum anderen kann ein Datum als Spalte des Typs DATE dargestellt werden.

Ist in einem Datumswert nur der Monat ohne Verweis auf einen Tag enthalten, wird als Standardeinstellung der erste Tag des Monats verwendet. Ist statt einem Monat oder einem Monat und Tag eine Saison enthalten, wird der erste Tag der Saison verwendet.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für Entrez-Datenquellen“ auf Seite 228

### Zugehörige Referenzen:

- „Angepasste Funktionen und Entrez-Abfragen“ auf Seite 230

## Entrez-Datenquelle - Beispielabfragen

In diesem Abschnitt sind einige Beispielabfragen für Entrez-Datenquellen aufgeführt.

### Vorgehensweise:

Verwenden Sie die nachstehenden Beispiele als Anleitung für die Ausführung von Abfragen.

### PubMed-Kurznamen:

Nachfolgend ist eine Abfrage mit einem einzelnen Abrufschlüssel für einen PubMed-Kurznamen dargestellt:

```
select PMID, ArticleTitle FROM pmarticles WHERE pmid = '12345';
```

Nachfolgend ist eine Abfrage mit gemischten Abrufschlüsseln für einen PubMed-Kurznamen dargestellt:

```
select PMID, ArticleTitle FROM pmarticles
WHERE pmid = '12345' OR MedlineID = '12346';
```

Nachfolgend ist eine Abfrage mit einer Funktion CONTAINS für einen PubMed-Kurznamen dargestellt:

```
select PMID, ArticleTitle FROM pmarticles
WHERE entrez.contains (ArticleTitle, 'granulation') = 1
AND entrez.contains (PubDate, '1992') = 1;
```

Nachfolgend ist eine Abfrage dargestellt, die nach den angegebenen Werten für 'AuthorList' und 'LanguageList' für einen PubMed-Kurznamen sucht:

```
select PMID, ArticleTitle FROM pmarticles
WHERE entrez.contains (AuthorList, 'Albarak') = 1
AND entrez.contains (LanguageList, 'eng')=1;
```

Nachfolgend ist eine Abfrage mit einem komplexen Vergleichselement für einen PubMed-Kurznamen dargestellt:

```
select PMID, ArticleTitle FROM pmarticles
WHERE entrez.contains (PublicationTypeList, 'Journal Article') = 1
AND entrez.contains (MedlineTA, 'sun')=1
OR entrez.contains (PersonalNameSubjectList, 'shine')=1;
```

#### **Nucleotide-Kurznamen:**

Nachfolgend ist eine Abfrage mit gemischten Abrufschlüsseln für einen Nucleotide-Kurznamen dargestellt:

```
select PrimaryAccession, LocusName, SeqLength from gbseq
WHERE PrimaryAccession in ('NM_000890', 'NC_003106');
```

Nachfolgend ist eine Abfrage dargestellt, die alle durchsuchbaren Felder nach einem Nucleotide-Kurznamen durchsucht:

```
select PrimaryAccession, substr(Definition,1,300), GI from gbseq
WHERE entrez.contains(AllText, 'abcde')=1;
```

Nachfolgend ist eine Abfrage dargestellt, die sämtlichen freien Text nach einem Nucleotide-Kurznamen durchsucht:

```
select * from gbseq WHERE entrez.contains(FreeText, 'abcde')=1;
```

Nachfolgend ist eine Abfrage dargestellt, die nach einer Definition für einen Nucleotide-Kurznamen sucht:

```
select PrimaryAccession, substr(Definition,1,300), version, GI from gbseq
WHERE entrez.contains(Definition, 'Sulfolobus tokodaii
AND complete genome') = 1;
```

Nachfolgend ist eine Abfrage dargestellt, die nach einem Schlüsselwort für einen Nucleotide-Kurznamen sucht:

```
select PrimaryAccession, substr(KeywordList,1,200), Segment from gbseq
WHERE entrez.contains(KeywordList, 'nkcc1 gene') = 1;
```



### Zugehörige Konzepte:

- „Relationale Vergleichselemente für den Entrez-Wrapper“ auf Seite 232
- „Ungültige WHERE-Klauseln für den Entrez-Wrapper“ auf Seite 232

### Zugehörige Tasks:

- „Registrieren von angepassten Funktionen für den Entrez-Wrapper“ auf Seite 223

## PubMed-Schematabellen

Das Schema PubMed definiert die Darstellung der Daten von einem Server des Typs 'PubMed' aus. Das Schema besteht aus mehreren, in Beziehung zueinander stehenden Kurznamen:

- PMArticles
- PMAccession
- PMChemical
- PMMeSHHeading
- PMComments
- PMArticleID

In den folgenden Tabellen werden Informationen zu den Spalten in den einzelnen Kurznamen aufgelistet. Die Spalte 'Kennungen' enthält die gültigen Kennungen für die Spalte. Wenn Sie eine Liste der gültigen Suchbefehle benötigen, besuchen Sie die nachstehende Website und suchen Sie dort die Verbindung (den Link) zu den Suchfeldbeschreibungen und -kennungen (Search Field Descriptions and Tags):

[www.ncbi.nlm.nih.gov/entrez/query/static/help/pmhelp.html](http://www.ncbi.nlm.nih.gov/entrez/query/static/help/pmhelp.html)

Sie können den Standarddatentyp für eine Spalte überschreiben, wenn Sie einen Kurznamen erstellen. Der Entrez-Wrapper unterstützt CLOB-Datentypen von bis zu 5 Megabyte Länge.

Sie können die Standardlänge für eine Spalte überschreiben, wenn Sie einen Kurznamen erstellen. Einige Spalten können große Datenmengen zurückgeben, z. B. die Spalte 'Abstract' im Kurznamen 'PMArticles'. Die Standardlänge für diese Spalte ist VARCHAR(32000). Zur Rückgabe der ersten 100 Byte der Spalte können Sie die Spalte mit dem Datentyp VARCHAR(100) definieren. Nur die ersten 100 Byte werden dann zurückgegeben.

### Kurzname PMArticles:

Die Spalten im Kurznamen PMArticles werden in folgender Tabelle beschrieben. Die Spalte F gibt Spalten an, die designierte Abrufschlüssel darstellen. Die Verwendung von Abrufschlüsseln kann die Abfrageverarbeitung beschleunigen.

Tabelle 48. PubMed-Kurzname PMArticles

Spaltenname	Datentyp	Beschreibung	Kennungen	Abrufschlüssel
PMID	VARCHAR(10) NOT NULL	Primärschlüsselspalte zum Verknüpfen des Kurznamens PMArticles mit den untergeordneten Kurznamen.	UID	Ja



Tabelle 48. PubMed-Kurzname PMArticles (Forts.)

Spaltenname	Datentyp	Beschreibung	Ken- nun- gen	Abruf- schlüs- sel
MedlineID	VARCHAR(10)	Medline ID	UID	Ja
Owner	VARCHAR(8) NOT NULL	Eigner des Publikations- eintrags. Die Werte werden vom NCBI definiert und können NLM, NASA, PIP, KIE, HSR, HMD, SIS, NOTNLM lauten. Ist kein Wert vorhanden, wird der Standardwert NLM ver- wendet.	keine	Nein
Status	VARCHAR(32) NOT NULL	Publikationsstatus wie vom NCBI definiert. Mögliche Werte sind: In-Process, Completed, Out-of-scope, PubMed-not_MEDLINE.	keine	Nein
DateCreated	DATE NOT NULL	Erstellungsdatum des Veröffentlichungseintrags	keine	Nein
DateCompleted	DATE	Fertigstellungsdatum des Veröffentlichungseintrags	keine	Nein
DateRevised	DATE	Revisionsdatum des Veröffentlichungseintrags	keine	Nein
ArticleTitle	VARCHAR(250) NOT NULL	Der Titel des Artikels	TI	Nein
Pagination	VARCHAR(32)	Die vollständige Paginierung des Artikels	keine	Nein
Abstract	VARCHAR(32000)	Die Zusammenfassung des Artikels	TIAB	Nein
Affiliation	VARCHAR(250)	Zuordnung und Adresse des ersten Autors	AD	Nein
AuthorList	VARCHAR(3200)	Liste mit Autoren; kanoni- siert	AU	Nein
LanguageList	VARCHAR(250) NOT NULL	Durch Semikolons unter- teilte Liste	LA	Nein
PublicationTypeList	VARCHAR(250) NOT NULL	Durch Semikolons unter- teilte Liste	PT	Nein
VernacularTitle	VARCHAR(250)	Der allgemeinsprachliche Titel des Artikels	keine	Nein
DateOfElectronic Publication	VARCHAR(32)	Das NCBI gibt für diese Spalte keine Struktur an.	keine	Nein
Country	VARCHAR(128)	Im Journal zitiertes Land oder Region der Veröffentli- chung	keine	Nein
MedlineTA	VARCHAR(250) NOT NULL	Die Abkürzung des Titels in Medline	TA	Nein
NlmUniqueID	VARCHAR(32)	Enthält MedlineCode, wenn NlmUniqueID nicht vor- handen	keine	Nein

Tabelle 48. PubMed-Kurzname PMArticles (Forts.)

Spaltenname	Datentyp	Beschreibung	Ken- nun- gen	Abruf- schlüs- sel
GeneSymbolList	VARCHAR(250)	Durch Semikolons unter- teilte Liste; seit 1996 nicht mehr in Gebrauch	keine	Nein
NumberOfReferences	INTEGER	Die Anzahl der bibliografi- schen Verweise für den Prüfartikel.	keine	Nein
PersonalNameSubjectList	VARCHAR(250)	Kanonisiert als durch Semi- kolons unterteilte Liste mit Namen	PS	Nein
KeywordList	VARCHAR(3200)	Durch Semikolons unter- teilte Liste	keine	Nein
SpaceFlightMissionList	VARCHAR(250)	Durch Semikolons unter- teilte Liste	keine	Nein
InvestigatorList	VARCHAR(250)	Kanonisiert als durch Semi- kolons unterteilte Liste mit Namen	keine	Nein
PublicationStatus	VARCHAR(32)	Der Status der Veröffentli- chung	keine	Nein
ProviderID	VARCHAR(32)	Die ID des Anbieters der Veröffentlichung	keine	Nein
CitationSubsetList	VARCHAR(250)	Durch Semikolons unter- teilte Liste	SB	Nein
AllFields	VARCHAR(1)	Pseudospalte; gibt stets NULL zurück	ALL	Nein
TextWords	VARCHAR(1)	Pseudospalte; gibt stets NULL zurück	TW	Nein
PubDate	DATE	Umfasst Journal- und Buch- publikationsdatum + Medline-Datum	DP	Nein
PubDateString	VARCHAR(32)	Umfasst Journal- und Buch- publikationsdatum + Medline-Datum	DP	Nein
Title	VARCHAR(250)	Buch- oder Journaltitel	TA	Nein
Journal_ISSN	CHAR(9)	Die ISSN des Journals	TA	Nein
Journal_Volume	VARCHAR(10)	Der Band des Journals	VI	Nein
Journal_Issue	VARCHAR(10)	Die Ausgabe des Journals	IP	Nein
Journal_Coden	VARCHAR(32)	Die Codenummer (coden) des Journals	keine	Nein
Journal_ISOAbbreviation	VARCHAR(32)	Die ISO-Abkürzung für das Journal	keine	Nein
Book_Publisher	VARCHAR(128)	Der Verleger des Buches	keine	Nein
Book_Authors	VARCHAR(250)	Kanonisiert als Liste mit anderen Autoren	keine	Nein
Book_CollectionTitle	VARCHAR(128)	Der Sammeltitle des Buches	keine	Nein
Book_Volume	VARCHAR(10)	Der Band des Buches	keine	Nein

### Kurzname PMAccession:

Die Spalten im Kurznamen PMAccession werden in folgender Tabelle beschrieben.

*Tabelle 49. PubMed-Kurzname PMAccession*

Spaltenname	Datentyp	Beschreibung	Ken- nun- gen
PMID	VARCHAR(10) NOT NULL	Der Schlüssel zur Verknüpfung eines untergeordneten Kurznamens PMAccession mit seinem übergeordneten Kurznamen.	keine
DataBankName	VARCHAR(250) NOT NULL	Der Name der Datenbank	SI
Accession	VARCHAR(32) NOT NULL	Die Zugriffsnummer	SI

### Kurzname PMChemical:

Die Spalten im Kurznamen PMChemical werden in folgender Tabelle beschrieben.

*Tabelle 50. PubMed-Kurzname PMChemical*

Spaltenname	Datentyp	Beschreibung	Ken- nun- gen
PMID	VARCHAR(10) NOT NULL	Der Schlüssel zur Verknüpfung eines untergeordneten Kurznamens PMChemical mit seinem übergeordneten Kurznamen.	keine
NameOfSubstance	VARCHAR(128) NOT NULL	Der Name der Substanz	NM
RegistryNumber	VARCHAR(32) NOT NULL	Kann eine CAS-Registrierungsnummer oder eine andere Registrierungsnummer sein	RN
CASRegistry	CHAR	Y oder N	keine

### Kurzname PMMeSHHeading:

Die Spalten im Kurznamen PMMeSHHeading werden in folgender Tabelle beschrieben.

*Tabelle 51. PubMed-Kurzname PMMeSHHeading*

Spaltenname	Datentyp	Beschreibung	Ken- nun- gen
PMID	VARCHAR(10) NOT NULL	Der Schlüssel zur Verknüpfung eines untergeordneten Kurznamens PMMeSHHeading mit seinem übergeordneten Kurznamen.	ID
DescriptorOrName	VARCHAR(128) NOT NULL	MeSH-Name oder -Deskriptor	MH <sup>1</sup>
DescriptorIsMajor	CHAR NOT NULL	Y, wenn der Deskriptor übergeordnet ist	keine

Tabelle 51. PubMed-Kurzname PMMeSHHeading (Forts.)

Spaltenname	Datentyp	Beschreibung	Ken- nun- gen
QualifierOrSubhead	VARCHAR(128)	MeSH-Qualifikationsmerkmal bzw. -Unterüberschrift	SH
QSIsmajor	CHAR	Y, wenn das Qualifikationsmerkmal oder das Subhead-Element übergeordnet ist	keine

**Anmerkungen:**

1. Ist das Vergleichselement "DescriptorIsMajor = Y" in der Abfrage enthalten, so ist der Suchbegriff MAJR.

**Kurzname PMComments:**

Die Spalten im Kurznamen PMComments werden in folgender Tabelle beschrieben.

Tabelle 52. PubMed-Kurzname PMComments

Spaltenname	Datentyp	Beschreibung	Ken- nun- gen
PMID	VARCHAR(10) NOT NULL	Der Schlüssel zur Verknüpfung eines untergeordneten Kurznamens PMComments mit seinem übergeordneten Kurznamen.	keine
RefSource	VARCHAR(128) NOT NULL	Die Verweisquelle	keine
Type	VARCHAR(32) NOT NULL	CommentOn, CommentIn, ErratumIn, ErratumFor, RepublishedFrom, RepublishedIn, RetractionOf, RetractionIn, UpdateIn, UpdateOf, SummaryForPatents, OriginalReportIn	keine
Note	VARCHAR(3200)	Anmerkungen	keine

**Kurzname PMArticleID:**

Die Spalten im Kurznamen PMArticleID werden in folgender Tabelle beschrieben.

Tabelle 53. PubMed-Kurzname PMArticleID

Spaltenname	Datentyp	Beschreibung	Ken- nun- gen
PMID	VARCHAR(10) NOT NULL	Der Schlüssel zur Verknüpfung eines untergeordneten Kurznamens PMArticleID mit seinem übergeordneten Kurznamen.	keine
ArticleID	VARCHAR(32) NOT NULL	Die ID des Artikels	keine
IdType	VARCHAR(8) NOT NULL	doi, pii, pmcpid, pmpid, sici, pubmed, medline, pmcid	keine

### Zugehörige Referenzen:

- „Vereinfachung von Schemadatenelementen“ auf Seite 232
- „Angepasste Funktionen und Entrez-Abfragen“ auf Seite 230
- „Nucleotide-Schematabellen“ auf Seite 240

## Nucleotide-Schematabellen

Das Nucleotide-Schema definiert die Darstellung der Daten von einem Server des Typs 'Nucleotide' aus. Das Schema besteht aus mehreren, in Beziehung zueinander stehenden Kurznamen:

- GBSeq
- GBReference
- GBFeatures
- GBIntervals
- GBQualifiers

In den folgenden Tabellen werden Informationen zu den Spalten in den jeweiligen Kurznamen aufgelistet. Die Spalte 'Qualifikationsmerkmal' enthält die gültigen Qualifikationsmerkmale für Suchoperationen für die Spalte. Wenn Sie eine Liste der gültigen Qualifikationsmerkmale für Suchoperationen benötigen, besuchen Sie die nachstehende Website, und suchen Sie dort die Verbindung (den Link) zu den Beschreibungen und Qualifikationsmerkmalen für Suchfelder (Search Field Descriptions and Qualifiers):

[www.ncbi.nlm.nih.gov/entrez/query/static/help/Summary\\_Matrices.html](http://www.ncbi.nlm.nih.gov/entrez/query/static/help/Summary_Matrices.html)

Sie können den Standarddatentyp für eine Spalte überschreiben, wenn Sie einen Kurznamen erstellen, z. B. den Standarddatentyp für die Spalte 'Sequence' im Kurznamen GBSeq: VARCHAR(32000). Sie können diesen Datentyp in CLOB (1 MB) ändern. Der Entrez-Wrapper unterstützt CLOB-Datentypen von bis zu 5 Megabyte Länge.

### GBSeq:

Die Spalten im Kurznamen GBSeq werden in folgender Tabelle beschrieben. Die Spalte F gibt Spalten an, die designierte Abrufschlüssel darstellen. Die Verwendung von Abrufschlüsseln kann die Abfrageverarbeitung beschleunigen.

Tabelle 54. Nucleotide-Kurzname GBSeq

Spaltenname	Datentyp	Beschreibung	Qualifikationsmerkmal	Abrufschlüssel
PrimaryAccession	VARCHAR(16) NOT NULL	Primäre Zugriffsnummer	PACC	Ja
SequenceKey	VARCHAR(32) NOT NULL	Primärschlüsselspalte zum Verknüpfen des Kurznamens GBSeq mit den untergeordneten Kurznamen.		Nein
LocusName	VARCHAR(16) NOT NULL	Der Name des Locus.	ACCN	Nein
SeqLength	INTEGER NOT NULL	Die Länge der Sequenz.	SLEN	Nein

Tabelle 54. Nucleotide-Kurzname GBSeq (Forts.)

Spaltenname	Datentyp	Beschreibung	Qualifikationsmerkmal	Abrufschlüssel
Strandedness	VARCHAR(32)	nicht festgelegt, einzelsträngig, doppelsträngig, gemischtsträngig	keines	Nein
MoleculeType	VARCHAR(16)	nucleic-acid, dna, rna, trna, rrna, mrna, urna, snrna, snorna, peptide	PROP	Nein
Topology	VARCHAR(16)	linear, rückbezüglich (zirkular)	keines	Nein
Division	CHAR(3) NOT NULL	Die GenBank-Division.	PROP	Nein
UpdateDate	DATE NOT NULL	Datum der letzten Aktualisierung.	MDAT	Nein
CreateDate	DATE NOT NULL	Erstellungsdatum des Eintrags.	keines	Nein
Definition	VARCHAR(7000) NOT NULL	Die Definitionszeile der Sequenz.	TITL	Nein
Version	INTEGER	Die Versions-ID der Sequenz.	keines	Nein
GI	VARCHAR(16)	Die Sequenz-ID für GenInfo (GI).	keines	Nein
KeywordList	VARCHAR(7000)	Durch Semikolons unterteilte Liste	KYWD	Nein
Segment	VARCHAR(250)	Das Segment.	keines	Nein
Source	VARCHAR(200) NOT NULL	Die Quelle.	ORGN	Nein
Organism	VARCHAR(7000) NOT NULL	Der Organismus.	ORGN	Nein
Taxonomy	VARCHAR(7000) NOT NULL	Die Taxonomie.	keines	Nein
Comment	VARCHAR(7000)	Kommentare	keines	Nein
Primary	VARCHAR(7000)	Die Primärspalte.	keines	Nein
SourceDB	VARCHAR(250)	Die Quelldatenbank.	keines	Nein
Sequence	VARCHAR(32000)	Die Sequenz.	keines	Nein
AllText	VARCHAR(1)	Pseudospalte, gibt stets NULL zurück	ALL	Nein
FreeText	VARCHAR(1)	Pseudospalte, gibt stets NULL zurück	WORD	Nein

I

### GBReference:

Die Spalten im Kurznamen GBReference werden in folgender Tabelle beschrieben.

*Tabelle 55. Nucleotide-Kurzname GBReference*

Spaltenname	Datentyp	Beschreibung	Qualifikationsmerkmal
SequenceKey	VARCHAR(32) NOT NULL	Der Schlüssel zur Verknüpfung eines untergeordneten Kurznamens GBReference mit seinem übergeordneten Kurznamen.	keines
ReferenceNum	INTEGER NOT NULL	Syntaktisch analysiert von GBReference_reference	keines
RangeLow	INTEGER NOT NULL	Niedrige Basis für Verweis (syntaktisch analysiert von GBReference_reference)	keines
RangeHigh	INTEGER NOT NULL	Hohe Basis für Verweis (syntaktisch analysiert von GBReference_reference)	keines
Authors	VARCHAR(3200)	Durch Semikolons unterteilte Liste mit Namen im GenBank-Format	AUTH
Consortium	VARCHAR(250)	Das Konsortium.	keines
Title	VARCHAR(250)	Der GenBank-Verweistitel.	WORD
Journal_Title	VARCHAR(250) NOT NULL	Der Titel des Journals.	JOUR
MedlineID	INTEGER	Die Medline-ID	keines
PubMedID	INTEGER	Die PubMed-ID	keines
Remarks	VARCHAR(3200)	Bemerkungen	keines

### GBFeatures:

Die Spalten im Kurznamen GBFeatures werden in folgender Tabelle beschrieben.

*Tabelle 56. Nucleotide-Kurzname GBFeatures*

Spaltenname	Datentyp	Beschreibung	Qualifikationsmerkmal
SequenceKey	VARCHAR(32) NOT NULL	Der Schlüssel zur Verknüpfung eines untergeordneten Kurznamens GBFeatures mit seinem übergeordneten Kurznamen.	keines
FeatureJoinKey	VARCHAR(32) NOT NULL	Primärschlüsselspalte zum Verknüpfen des Kurznamens GBFeatures mit den untergeordneten Kurznamen.	keines
FeatureKey	VARCHAR(20) NOT NULL		FKEY
FeatureLocation	VARCHAR(200) NOT NULL		keines

### GBIntervals:

Die Spalten im Kurznamen GBIntervals werden in folgender Tabelle beschrieben.

*Tabelle 57. Nucleotide-Kurzname GBIntervals*

Spaltenname	Datentyp	Beschreibung	Qualifikationsmerkmal
FeatureJoinKey	VARCHAR(32) NOT NULL	Der Schlüssel zur Verknüpfung eines untergeordneten Kurznamens GBIntervals mit seinem übergeordneten Kurznamen.	keines
IntervalFrom	INTEGER		keines
IntervalTo	INTEGER		keines
IntervalPoint	INTEGER		keines
IntervalAccession	VARCHAR(32) NOT NULL		keines

### GBQualifiers:

Die Spalten im Kurznamen GBQualifiers werden in folgender Tabelle beschrieben.

*Tabelle 58. Nucleotide-Kurzname GBQualifiers*

Spaltenname	Datentyp	Beschreibung	Qualifikationsmerkmal
FeatureJoinKey	VARCHAR(32) NOT NULL	Der Schlüssel zur Verknüpfung eines untergeordneten Kurznamens GBQualifiers mit seinem übergeordneten Kurznamen.	keines
QualifierName	VARCHAR(50)	Name des Qualifikationsmerkmals	keines
QualifierValue	VARCHAR(32000)	Wert des Qualifikationsmerkmals	keines

### Zugehörige Referenzen:

- „Vereinfachung von Schemadatenelementen“ auf Seite 232
- „PubMed-Schematabellen“ auf Seite 235
- „Angepasste Funktionen und Entrez-Abfragen“ auf Seite 230



---

## Nachrichten für den Entrez-Wrapper

In diesem Abschnitt werden Nachrichten beschrieben, die beim Arbeiten mit dem Wrapper für Entrez auftreten können. Wenden Sie sich bei Nachrichten, die nicht in dieser Tabelle bzw. in Band 1 oder Band 2 der Dokumentation zu Fehlermeldungen (Message Reference: Volume 1, Message Reference: Volume 2) enthalten sind, an die IBM Softwareunterstützung.

*Tabelle 59. Vom Entrez-Wrapper ausgegebene Nachrichten*

Fehlercode	Nachricht	Erläuterung
SQL0142N	Die SQL-Anweisung wird nicht unterstützt.	Ein ungültiger Abfragetyp wurde an den Wrapper übergeben. Überprüfen Sie, ob die ausgegebene SQL-Anweisung von diesem Wrapper unterstützt wird.
SQL0204N	"<name>" ist ein undefinierter Name.	Der angegebene Name ist ungültig. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL0405N	Das numerische Literal "<literal>" ist nicht gültig, da sein Wert außerhalb des zulässigen Bereichs liegt.	Eine Spalte in den abgerufenen XML-Daten oder ein Vergleichselement in einer SQL-Anweisung enthält einen Wert, der außerhalb des zulässigen Wertebereichs für den Datentyp liegt. Überprüfen Sie den Datentyp für diese Spalte sowie die Spalte in der Datenquelle, oder definieren Sie die Spalte erneut mit einem passenderen Typ.
SQL0408N	Ein Wert ist mit dem Datentyp seines Zuordnungsziels nicht kompatibel. Der Name des Ziels ist "<zielname>".	Eine Spalte in den XML-Daten enthält Spalten, die für diesen Datentyp ungültig sind. Überprüfen Sie den Datentyp für diese Spalte sowie die Spalte in der Datenquelle, oder definieren Sie die Spalte erneut mit einem passenderen Typ.
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Datenbankprototyp konnte nicht gefunden werden.")	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.

Tabelle 59. Vom Entrez-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Keine zu entpackenden Daten vorhanden.")	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Fehler beim Erstellen eines Wrapperobjekts".)	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Ungültiger Ausdruckstyp.")	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Kurzname konnte nicht gefunden werden.")	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Fehler bei Speicherzuordnung.")	Es ist keine ausreichende Speicherkapazität zum Verarbeiten der Zuordnungsanforderung innerhalb des Wrappers vorhanden.
SQL1816N	Der Wrapper "<wrappername>" kann nicht für den Zugriff auf die "Version" der Datenquelle ("<servertyp>", "<serverversion>") verwendet werden, die Sie für den Server mit zusammengeschlossenen Datenbanken zu definieren versuchen.	Ein Wert der Klausel VERSION der Anweisung CREATE SERVER ist ungültig.

Tabelle 59. Vom Entrez-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1816N	Der Wrapper " <code>&lt;wrappername&gt;</code> " kann nicht für den Zugriff auf den "Typ" der Datenquelle (" <code>&lt;servertyp&gt;</code> ", " <code>&lt;serverversion&gt;</code> ") verwendet werden, die Sie für den Server mit zusammengeschlossenen Datenbanken zu definieren versuchen.	Ein Wert der Klausel TYPE der Anweisung CREATE SERVER ist ungültig.
SQL1817N	Die Anweisung CREATE SERVER gibt den "Typ" der Datenquelle nicht an, die Sie für die zusammengeschlossene Datenbank definieren möchten.	Die Klausel TYPE der Anweisung CREATE SERVER ist erforderlich, wurde jedoch nicht angegeben.
SQL1822N	Es wurde ein unerwarteter Fehlercode "900" von der Datenquelle "Entrez-Wrapper" empfangen. Zugeordneter Text und Token sind "Übergeordneter Kurzname ist nicht definiert".	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1823N	Für den Datentyp " <code>&lt;datentyp&gt;</code> " gibt es vom Server " <code>&lt;servername&gt;</code> " keine Datentypzuordnung.	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1881N	" <code>&lt;optionsname&gt;</code> " ist keine gültige Option des Typs " <code>&lt;optionstyp&gt;</code> " für " <code>&lt;optionsname&gt;</code> ".	Die angegebene Option ist ungültig. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1882N	Die Option " <code>&lt;optionsname&gt;</code> " der Art " <code>&lt;optionstyp&gt;</code> " kann nicht auf " <code>&lt;optionswert&gt;</code> " für " <code>&lt;optionsname&gt;</code> " eingestellt werden.	Der angegebene Wert ist für diese Option ungültig. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1883N	" <code>&lt;optionsname&gt;</code> " ist eine erforderliche Option der Art " <code>&lt;optionstyp&gt;</code> " für " <code>&lt;optionsname&gt;</code> ".	Diese Option ist für das Objekt erforderlich, wurde jedoch nicht angegeben. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1884N	Sie haben "FOREIGN_KEY" (eine Option der Art "COLUMN") mehrfach angegeben.	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1884N	Sie haben "PRIMARY_KEY" (eine "COLUMN"-Option) mehrmals angegeben.	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.

Tabelle 59. Vom Entrez-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Serverversion konnte nicht geändert werden."	Die Version eines Servers kann nicht durch Ausgeben der Anweisung ALTER SERVER geändert werden. Die Erstellung eines neuen Servers mit der neuen Version ist erforderlich.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Ungültiger PARENT-Kurzname."	Der Kurzname, auf den in einer Kurznamenoption PARENT verwiesen wird, ist für den aktuellen Kurznamen ungültig.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Ungültiger Spaltenname."	Ein in der Anweisung CREATE NICKNAME angegebener Spaltenname stimmt mit keiner der für den Kurznamen in Frage kommenden Spalten überein.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Abrufschlüssel können nicht mit AND verbunden werden."	Mehrere Verweise auf einen Abrufschlüssel, z. B. auf die Spalte PMID des Kurznamens PMArticles, wurden miteinander verbunden. Beispiel: "PMID = 12346 AND PMID = 12348". Vergleichselemente für Abrufschlüssel können nur mit OR zugeordnet werden.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Funktionen SEARCH_TERM und CONTAINS wurden gemischt."	Die Funktionen SEARCH_TERM und CONTAINS können in einer Abfrage nicht kombiniert werden. Pro Abfrage ist nur eine einzige Funktion SEARCH_TERM erlaubt.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Ungültiges erstes Argument in der Funktion."	Das erste Argument für eine Funktion SEARCH_TERM oder CONTAINS war ungültig. Dieses Argument muss ein Verweis auf eine Spalte sein.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Ungültiges zweites Argument in der Funktion."	Das zweite Argument für eine Funktion SEARCH_TERM oder CONTAINS war ungültig. Dieses Argument muss ein Zeichenfolgeliteral, eine Hostvariable oder ein Verweis auf eine Spalte sein.

Tabelle 59. Vom Entrez-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Spalte ohne Kennung in CONTAINS-Funktion".	Das erste Argument für die Funktion CONTAINS war ungültig. Dieses Argument muss ein Verweis auf eine Spalte mit Kennung sein.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "Ungültige Funktion".	Dies ist ein interner Fehler. Wenden Sie sich an die IBM Softwareunterstützung.

**Zugehörige Konzepte:**

- „Nachrichten - Einführung“ in *Fehlernachrichten Band 1*

**Zugehörige Referenzen:**

- „SQLSTATE-Nachrichten“ in *Fehlernachrichten Band 2*

## Kapitel 12. Konfigurieren des Zugriffs auf Excel-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Excel-Datenquellen gespeichert sind. Der Zugriff auf Excel-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält folgende Informationen:

- Erläuterungen zu Excel.
- Liste der auszuführenden Tasks.
- Beispiele zu den benötigten SQL-Anweisungen.
- Liste der dem Excel-Wrapper zugeordneten Fehlermeldungen.

### Was ist Excel?

Ein Excel-Arbeitsblatt bzw. eine Excel-Arbeitsmappe ist eine Datei, die mit der Anwendung Microsoft® Excel erstellt wurde und eine Dateierweiterung 'xls' hat. DB2® Information Integrator unterstützt Arbeitsblätter von Excel 97, Excel 2000 und Excel 2002. In Abb. 23 ist dargestellt, wie der Excel-Wrapper die Arbeitsblätter mit Ihrem System zusammengeschlossener Datenbanken verbindet.

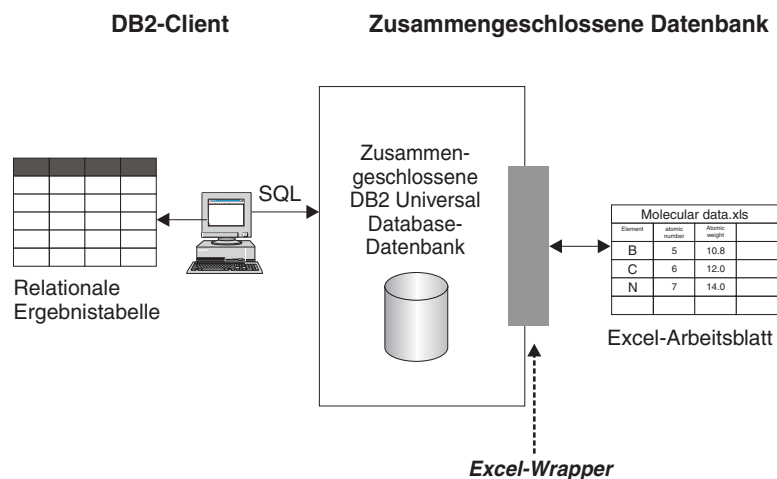


Abbildung 23. Funktionsweise des Excel-Wrappers

Der Excel-Wrapper verwendet die Anweisung CREATE NICKNAME, um die Spalten in Ihrer Excel-Tabelle (Arbeitsblatt) zu den Spalten in Ihrem DB2 UDB-System zusammengeschlossener Datenbanken zuzuordnen. Tabelle 60 zeigt Daten aus einem Beispielarbeitsblatt, die in einer Datei namens Compound\_Master.xls gespeichert sind.

Tabelle 60. Beispieltabelle für Compound\_Master.xls

	A	B	C	D
1	COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
2	compound_A	1.23	367	tested
3	compound_G		210	

Tabelle 60. Beispieltabelle für Compound\_Master.xls (Forts.)

	A	B	C	D
4	compound_F	0.000425536	174	tested
5	compound_Y	1.00256		tested
6	compound_Q		1024	
7	compound_B	33.5362		
8	compound_S	0.96723	67	tested
9	compound_O	1.2		tested

Diese Informationen stehen normalerweise nicht über die SQL-Standardbefehle zur Verfügung. Wenn der Excel-Wrapper installiert und registriert ist, können Sie auf diese Informationen zugreifen, als ob es sich um eine relationale Standarddatenquelle handeln würde. Wenn Sie beispielsweise die Daten zu allen Bestandteilen (compound) benötigen, deren Molekularzahl (molecular\_count) größer als 100 ist, müssen Sie die folgende SQL-Abfrage ausführen:

```
SELECT * FROM compound_master WHERE mol_count > 100
```

Die Abfrageergebnisse sind in Tabelle 61 dargestellt.

Tabelle 61. Abfrageergebnisse

COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
compound_A	1.23	367	tested
compound_G		210	
compound_F	0.000425536	174	tested
compound_Q		1024	

#### Zugehörige Konzepte:

- „Methoden für den Zugriff auf Excel-Daten“ auf Seite 30

#### Zugehörige Tasks:

- „Hinzufügen von Excel-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 250

---

## Hinzufügen von Excel zu einem Server mit zusammengeschlossenen Datenbanken

### Hinzufügen von Excel-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Excel-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken auf Excel-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

**Voraussetzungen:**

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.
- Excel-Arbeitsblätter, die korrekt strukturiert sind, sodass der Wrapper auf die Daten zugreifen kann.

**Vorgehensweise:**

Führen Sie folgende Schritte aus, um die Excel-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Registrieren Sie den Wrapper.
2. Registrieren Sie die Serverdefinition.
3. Registrieren Sie Kurznamen für die Excel-Arbeitsblätter.

**Zugehörige Konzepte:**

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Methoden für den Zugriff auf Excel-Daten“ auf Seite 30
- „Was ist Excel?“ auf Seite 249
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

**Zugehörige Tasks:**

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Registrieren des Excel-Wrappers“ auf Seite 251
- „Registrieren des Servers für eine Excel-Datenquelle“ auf Seite 253
- „Registrieren von Kurznamen für Excel-Datenquellen“ auf Seite 253
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43

**Zugehörige Referenzen:**

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Auswählen des korrekten Wrappers“ auf Seite 29

## Registrieren des Excel-Wrappers

Die Registrierung des Excel-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von Excel-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf Excel-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.



### Einschränkungen:

- Der Excel-Wrapper ist nur für Microsoft Windows-Betriebssysteme verfügbar, die DB2 Universal Database Enterprise Server Edition unterstützen.
- Die Excel-Anwendung muss auf dem Server installiert sein, auf dem auch DB2 Information Integrator installiert ist, bevor der Excel-Wrapper genutzt werden kann.
- Durchgriffssitzungen werden nicht unterstützt.

### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Bibliotheksdatei des Wrappers aus.

Um beispielsweise einen Wrapper mit dem Namen excel\_wrapper zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER excel_wrapper LIBRARY 'db2lsxls.dll';
```

Sie müssen die Bibliotheksdatei des Wrappers db2lsxls.dll in der Anweisung CREATE WRAPPER angeben.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinition für den Excel-Wrapper.

### Zugehörige Tasks:

- „Registrieren des Servers für eine Excel-Datenquelle“ auf Seite 253

### Zugehörige Referenzen:

- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien für den Excel-Wrapper

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den Excel-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, wird diese Bibliotheksdatei zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 62. Position und Dateiname der Excel-Wrapperbibliothek*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
Windows	%DB2PATH%\bin	db2lsxls.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist C:\Program Files\IBM\SQLLIB.

### Zugehörige Tasks:

- „Registrieren des Excel-Wrappers“ auf Seite 251

## Registrieren des Servers für eine Excel-Datenquelle

Die Registrierung des Servers für eine Excel-Datenquelle ist Teil der umfangreicheren Task des Hinzufügens von Excel zu einem System zusammenschlossener Datenbanken. Nach der Registrierung des Wrappers muss ein entsprechender Server registriert werden.

Für Excel wird eine Serverdefinition erstellt, da die Hierarchie von Objekten zusammenschlossener Datenbanken erfordert, dass Datenquellendateien (die durch Kurznamen identifiziert werden) einem spezifischen Serverobjekt zugeordnet werden.

### Vorgehensweise::

Um den Excel-Server für das System zusammenschlossener Datenbanken zu registrieren, verwenden Sie die Anweisung CREATE SERVER.

Angenommen, Sie wollen für eine Arbeitsmappe ein Serverobjekt namens `biochem_lab` erstellen, das biochemische Daten enthält. Das Serverobjekt muss dem Excel-Wrapper zugeordnet werden, den Sie mit Hilfe der Anweisung CREATE WRAPPER registriert haben. Die entsprechende Anweisung CREATE SERVER zum Registrieren dieses Serverobjekts lautet wie folgt:

```
CREATE SERVER biochem_lab WRAPPER Excel_2000-Wrapper;
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für Excel-Datenquellen.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für Excel-Datenquellen“ auf Seite 253

### Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- „CREATE SERVER-Anweisungsargumente - Excel-Wrapper“ auf Seite 609

## Registrieren von Kurznamen für Excel-Datenquellen

Das Registrieren der Kurznamen für Excel-Datenquellen ist Teil der umfangreicheren Task des Hinzufügens von Excel zu einem System zusammenschlossener Datenbanken. Nach der Registrierung eines Servers muss ein entsprechender Kurzname registriert werden. Kurznamen werden in Abfragen zum Verweisen auf Excel-Datenquelle verwendet.

### Vorgehensweise::

Wenn Sie die Excel-Datenquelle relationalen Tabellen zuordnen möchten, müssen Sie mit Hilfe der Anweisung CREATE NICKNAME einen Kurznamen erstellen.

Mit der Anweisung im folgenden Beispiel wird der Kurzname 'Compounds' aus der Excel-Tabellendatei `CompoundMaster.xls` erstellt. Die Datei enthält drei Datenspalten, die für das System zusammenschlossener Datenbanken als `Compound_ID`, `CompoundName` und `MolWeight` definiert sind.

```
CREATE NICKNAME Compounds (  
    Compound_ID INTEGER,  
    CompoundName VARCHAR(50),
```

```

        MolWeight FLOAT)
FOR SERVER biochem_lab
    OPTIONS (FILE_PATH 'C:\My Documents\CompoundMaster.xls',
            RANGE 'B2:D5');

```

Dies war die letzte Task in dieser Tasksequenz.

#### Zugehörige Tasks:

- „Hinzufügen von Excel-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 250
- „Angabe von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75

#### Zugehörige Referenzen:

- „Excel-Datenquelle - Beispielabfragen“ auf Seite 254
- „CREATE NICKNAME-Anweisungssyntax - Excel-Wrapper“ auf Seite 609

---

## Excel-Datenquelle - Beispielabfragen

In diesem Abschnitt wird eine Reihe von Beispielabfragen für Excel-Tabellen aufgeführt, für die der Beispielkurzname Compounds verwendet wird.

Um Abfragen auszuführen, verwenden Sie den Kurznamen und die definierten Kurznamenspalten in Ihren SQL-Anweisungen in derselben Weise wie reguläre Tabellennamen und Tabellenspalten.

Mit der folgenden Abfrage werden alle Bestandteil-IDs (compound\_ID) angezeigt, für die das Molekulargewicht (MolWeight) größer als 200 ist:

```

SELECT compound_ID
FROM Compounds
WHERE MolWeight > 200;

```

Mit der folgenden Abfrage werden alle Datensätze angezeigt, für die der Name des Bestandteils (compound\_name) oder das molekulare Gewicht (weight) null ist:

```

SELECT *
FROM Compounds
WHERE CompoundName IS NULL
OR MolWeight IS NULL;

```

Mit der folgenden Abfrage werden alle Datensätze angezeigt, für die der Name des Bestandteils (compound\_name) die Zeichenfolge ase enthält und das Molekulargewicht (MolWeight) größer-gleich 300 ist:

```

SELECT *
FROM Compounds
WHERE CompoundName LIKE '%ase%'
AND MolWeight >=300;

```

#### Zugehörige Referenzen:

- „Documentum-Datenquelle - Beispielabfragen“ auf Seite 212
- „Excel-Datenquelle - Beispielszenario“ auf Seite 255

## Excel-Datenquelle - Beispielszenario

In diesem Abschnitt wird eine Beispielimplementierung des Wrappers 'Excel\_2000' dargestellt, der auf ein Excel 2000-Arbeitsblatt im Verzeichnis C:\Data zugreift. In diesem Szenario werden der Wrapper, ein Server und ein Kurzname für den Zugriff auf das Arbeitsblatt registriert. Die in diesem Szenario dargestellten Anweisungen werden über die DB2-Befehlszeile eingegeben. Nach der Registrierung des Wrappers können Abfragen für das Arbeitsblatt ausgeführt werden.

Das Szenario beginnt mit einem Arbeitsblatt für Bestandteile; dieses heißt Compound\_Master.xls und besteht aus 4 Spalten und 9 Zeilen. Der vollständig qualifizierte Pfadname für die Datei lautet C:\Data\Compound\_Master.xls. Der Inhalt ist in Tabelle 63 dargestellt.

Tabelle 63. Beispielarbeitsblatt Compound\_Master.xls

	A	B	C	D
1	COMPOUND_NAME	WEIGHT	MOL_COUNT	WAS_TESTED
2	compound_A	1.23	367	tested
3	compound_G		210	
4	compound_F	0.000425536	174	tested
5	compound_Y	1.00256		tested
6	compound_Q		1024	
7	compound_B	33.5362		
8	compound_S	0.96723	67	tested
9	compound_O	1.2		tested

### Vorgehensweise:

Gehen Sie wie folgt vor, um mit Hilfe des Excel-Wrappers auf ein Excel 2000-Arbeitsblatt zuzugreifen:

1. Registrieren Sie den Excel\_2000-Wrapper:  
`db2 => CREATE WRAPPER Excel_2000 LIBRARY 'db21sx1s.dll'`
2. Registrieren Sie den Server:  
`db2 => CREATE SERVER biochem_lab WRAPPER Excel_2000`
3. Registrieren Sie einen Kurznamen, der auf das Excel-Arbeitsblatt verweist:  
`db2 => CREATE NICKNAME Compound_Master (compound_name VARCHAR(40),  
weight FLOAT, mol_count INTEGER, was_tested VARCHAR(20))  
FOR biochem_lab  
OPTIONS ( FILE_PATH 'C:\Data\Compound_Master.xls')`

Der Registrierungsprozess ist abgeschlossen. Die Excel-Datenquelle ist nun Teil des Systems mit zusammengeschlossenen Datenbanken und kann in SQL-Abfragen verwendet werden.

In den folgenden Beispielen sind SQL-Beispielabfragen und -Ergebnisse dargestellt, die anhand der Excel-Datenquelle abgerufen wurden.

- SQL-Beispielabfrage: "Alle Daten für Bestandteile abrufen, für die mol\_count größer als 100 ist."  
`SELECT * FROM compound_master WHERE mol_count > 100`  
Ergebnis: Alle Felder für die Zeilen 2, 3, 4, 6 und 8.

- SQL-Beispielabfrage: "Die Namen von Bestandteilen (compound\_name) und Molekularzahlen (mol\_count) für alle Bestandteile abrufen, für die mol\_count noch nicht ermittelt wurde."

```
SELECT compound_name, mol_count FROM compound_master
WHERE mol_count IS NULL
```

Ergebnis: Die Felder compound\_name und mol\_count der Zeilen 5, 7 und 10 im Arbeitsblatt.

- SQL-Beispielabfrage: "Die Anzahl der Bestandteile angeben, für noch kein Test durchgeführt wurde und deren Gewicht größer als 1 ist."

```
SELECT count(*) FROM compound_master
WHERE was_tested IS NULL AND weight > 1
```

Ergebnis: Die Satzzählung 1, die die Zeile 7 der Tabelle angibt, die die Kriterien erfüllt.

- SQL-Beispielabfrage: "Die Namen von Bestandteilen (compound\_name) und Molekularzahlen (mol\_count) für alle Bestandteile abrufen, für die mol\_count ermittelt wurde und die unter dem Durchschnittswert für mol\_count liegen."

```
SELECT compound_name, mol_count
FROM compound_master
WHERE mol_count IS NOT NULL
AND mol_count < (SELECT AVG(mol_count) FROM compound_master
WHERE mol_count IS NOT NULL AND was_tested IS NOT NULL)
```

Die Unterabfrage gibt den Durchschnitt 368 an die Hauptabfrage zurück, die anschließend Tabelle 64 zurückgibt:

*Tabelle 64. Abfrageergebnisse*

COMPOUND_NAME	MOL_COUNT
compound_A	367
compound_G	210
compound_F	174
compound_S	67

#### **Zugehörige Tasks:**

- „Hinzufügen von Excel-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 250

#### **Zugehörige Referenzen:**

- „Excel-Datenquelle - Beispielabfragen“ auf Seite 254

---

## **Modell für die Dateizugriffssteuerung für den Excel-Wrapper**

Das Datenbankverwaltungssystem greift auf Excel-Dateien mit der Berechtigung LOG ON AS als Eigenschaft des DB2-Datenbankservices zu. Diese Einstellung kann auf der Eigenschaftsseite LOG ON für das DB2-Exemplar angezeigt werden. Auf die Eigenschaftsseite kann über 'Dienste' in der Windows NT-Systemsteuerung zugegriffen werden.

#### **Zugehörige Referenzen:**

- „Modell der Dateizugriffssteuerung für den Wrapper für Dateien mit Tabellenstruktur“ auf Seite 410
- „Zugriffssteuerung für den Documentum-Wrapper“ auf Seite 214

---

## Nachrichten für den Excel-Wrapper

In diesem Abschnitt sind Nachrichten und die zugehörigen Beschreibungen aufgeführt, die bei der Verwendung des Wrappers für Excel angezeigt werden können.

*Tabelle 65. Vom Excel-Wrapper ausgegebene Nachrichten*

<b>Fehlercode</b>	<b>Nachricht</b>	<b>Erläuterung</b>
SQL1817N	Die Anweisung CREATE SERVER gibt die "VERSION" der Datenquelle nicht an, die Sie für die zusammengeschlossene Datenbank definieren möchten.	Der Parameter VERSION wurde in der Anweisung CREATE SERVER nicht angegeben. Korrigieren Sie die SQL-Anweisung, und setzen Sie sie erneut ab.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1000.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Speicherzuordnungsfehler".	Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1001.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Unbekannte Option".	Die in der DLL-Anweisung angegebene Option wird nicht unterstützt. Korrigieren Sie die SQL-Anweisung, und setzen Sie sie erneut ab.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1002.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Erstellen des DELTA-Objekts fehlgeschlagen".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1100.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Wrapperoptionen werden nicht unterstützt".	Optionen für Wrapper (OPTIONS) werden von diesem Wrapper nicht unterstützt. Korrigieren Sie die SQL-Anweisung, und setzen Sie sie erneut ab.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1200.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "<option> ist eine nicht unterstützte Serveroption".	Die angegebene Option wird von diesem Wrapper nicht unterstützt. Korrigieren Sie die SQL-Anweisung, und setzen Sie sie erneut ab.

Tabelle 65. Vom Excel-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1201.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Abrufen des Servernamens".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1209.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Umsetzen von VARCHAR-Daten".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1211.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Umsetzen von INTEGER-Daten".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1212.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Umsetzen von FLOAT-Daten".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1400.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "<option> ist eine nicht unterstützte Benutzeroption".	Die angegebene Option wird von diesem Wrapper nicht unterstützt. Korrigieren Sie die SQL-Anweisung, und setzen Sie sie erneut ab.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1401.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Erstellung des Deltaobjekts USER fehlgeschlagen".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1500.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "<option> ist eine nicht unterstützte Kurznamenoption".	Die angegebene Option wird von diesem Wrapper nicht unterstützt. Korrigieren Sie die SQL-Anweisung, und setzen Sie sie erneut ab.

Tabelle 65. Vom Excel-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1501.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Erforderliche Option PATH nicht angegeben".	Die Option PATH ist erforderlich, um den KURZNAMEN zu registrieren. Korrigieren Sie die SQL-Anweisung, und setzen Sie sie erneut ab.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1502.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Erstellung des Deltaobjekts NICKNAME fehlgeschlagen".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1503.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Abrufen des Spaltentyps für Kurznamen".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1504.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Abrufen des Spaltentyps für Kurznamen".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1505.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token wurden von Datenquelle "Excel-Wrapper" empfangen.	Der angegebene <datentyp> wird von diesem Wrapper nicht unterstützt. Korrigieren Sie die SQL-Anweisung, und setzen Sie sie erneut ab.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1506.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Abrufen der Spalteninformationen für Kurznamen".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.



Tabelle 65. Vom Excel-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1507.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Option <option> kann nicht weggelassen werden".	Die angegebene Option kann nicht weggelassen werden, da es sich um eine erforderliche Option handelt.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1508.VANI" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Spaltennamen können nicht geändert werden".	Das Ändern von Spaltennamen ist im Excel-Wrapper nicht zulässig.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1509.VCTS" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Spalteninformationen wurden nicht gefunden".	Die Spalteninformationen wurden nicht gefunden.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1701.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler bei der SQL-Syntaxanalyse".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1702.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Zugriff auf das Kurznamenobjekt".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1703.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler bei der Erstellung eines Datenspeicherbereichs".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1704.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Herstellen einer Verbindung zwischen SQL und Kurznamen-daten".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.

Tabelle 65. Vom Excel-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1705.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Starten der Excel-Anwendung fehlgeschlagen".	Das Starten der Excel-Anwendung ist fehlgeschlagen. Stellen Sie sicher, dass Excel auf dem System installiert ist und für die korrekte Version des Wrappers registriert wurde. Überprüfen Sie die Eigenschaft LOG ON AS für das DB2-Exemplar unter 'Dienste' in der Windows NT-Systemsteuerung. Auf die Excel-Anwendung wird über diese Berechtigung zugegriffen. Stellen Sie sicher, dass der jeweilige Benutzer über die entsprechenden Zugriffsberechtigungen verfügt bzw. geben Sie für diese Eigenschaft einen berechtigten Benutzereintrag an. Starten Sie anschließend DB2 erneut und führen Sie die SQL-Abfrage erneut aus.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1706.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Öffnen der Quellentabelle".	Beim Öffnen der Tabelle, auf die durch den Kurznamen in der SQL-Abfrage verwiesen wird, trat ein Fehler auf. Stellen Sie sicher, dass sich die Datei in dem Pfad befindet, der in der Anweisung CREATE NICKNAME bei der Registrierung angegeben wurde.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1707.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Zugriff auf den DL-Ausgabespeicherbereich".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1708.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Beenden der Excel-Anwendung fehlgeschlagen".	Ein interner Programmfehler ist aufgetreten. Tritt dieser Fehler nach wiederholten Abfragen weiterhin auf, die IBM Softwareunterstützung benachrichtigen.
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1711.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Fehler beim Abrufen; Daten-/Spaltentyp stimmen möglicherweise nicht überein".	Die bei der SQL-Abfrage abgerufenen Daten hatten nicht den Datentyp, der bei der Registrierung des Kurznamens angegeben wurde. Korrigieren Sie die Daten in der Quellentabelle oder korrigieren Sie den registrierten Datentyp im Kurznamen. Kann der Fehler dadurch nicht behoben werden, wenden Sie sich an die IBM Softwareunterstützung.

Tabelle 65. Vom Excel-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "-1900.<interner_programmcode>" von der Datenquelle "Excel-Wrapper" empfangen. Zugeordneter Text und Token sind "Speicherzuordnungsfehler".	Ein interner Programmfehler ist aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.

**Zugehörige Konzepte:**

- „Nachrichten - Einführung“ in *Fehlernachrichten Band 1*

**Zugehörige Referenzen:**

- „SQLSTATE-Nachrichten“ in *Fehlernachrichten Band 2*

---

## Kapitel 13. Konfigurieren des Zugriffs auf Extended Search-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Datenquellen von IBM Lotus Extended Search gespeichert sind. Der Zugriff auf Extended Search-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält folgende Informationen:

- Erläuterungen zu Extended Search.
- Liste der auszuführenden Tasks.
- Beispiele zu den benötigten SQL-Anweisungen.
- Liste der dem Extended Search-Wrapper zugeordneten Fehlernachrichten.

---

### Was ist Extended Search?

Das Produkt Extended Search ist ein mehrschichtiges Client/Server-System, das umfangreiche Such- und Abruffunktionen zur Verfügung stellt. Bei Extended Search können Sie eine einzelne Anforderung eingeben und gleichzeitig potenziell Tausende von Datenrepositorys sowie das Internet durchsuchen. Diese Repositorys können die unterschiedlichsten Inhalte und Strukturen haben, und sind unter Umständen über die ganze Welt verstreut.

Extended Search unterstützt die verteilte und heterogene Suche von strukturierten und unstrukturierten Daten über einen einzelnen Zugriffspunkt. Es nutzt den aktuellen Aufwand für Ihre Datenverwaltung und führt vollständig die zum gleichzeitigen Zugriff auf mehrere verschiedenen Datenquellen erforderliche Logistik aus.

Extended Search verwendet seine Abfragesprache GQL (Generalized Query Language) als gemeinsame Suchsyntax und setzt jede Suchanfrage intern in die nativen Suchsprachen der zu durchsuchenden Datenquellen um. Es verwendet auch native Methoden für Quellen, die unabhängig von der Position einer Quelle Informationen suchen und abrufen.

Informationen zum Installieren des Extended Search-Servers, Konfigurieren der Suchdomäne und die Verwendung von GQL finden Sie in der Produktdokumentation zu Extended Search. Folgende Dokumente sind auf der Ressourcen-seite der IBM® Lotus® Extended Search-Website verfügbar:

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

#### *Extended Search Allgemeine Informationen*

Beschreibt die Komponenten in einem Extended Search-System sowie deren Interaktion untereinander und mit den Back-End-Datensystemen.

#### *Extended Search Installation*

Definiert die Systemvoraussetzungen und bietet Instruktionen zur Produktinstallation und zur Prüfung des Installationsprozesses.

### *Extended Search Administration*

Bietet Instruktionen zum Hinzufügen von Datenquellen zur Suchdomäne, zum Konfigurieren von Suchfeldern und zur Verwendung von Beispielsuchanwendungen zum Abfragen von Extended Search-Datenquellen.

### *Extended Search Programming*

Erläutert die Tools für die Anwendungsentwicklung, die Sie zur Erweiterung der Unterstützung für die Suche für Datenquellen verwenden können, die in der Standardkonfiguration des Produkts nicht unterstützt werden. Enthält eine Beschreibung der Abfragesprache GQL von Extended Search.

## **Extended Search-Datenquellen**

Mit Extended Search können Sie folgende Typen von Datenquellen durchsuchen:

- Viele vielfach eingesetzte Websites für Suchen und News. Sie können für die Site für Suchen in Ihrem Intranet bzw. andere interne oder externe Sites für Suchen auf einfache Weise Unterstützung hinzufügen.
- Mailsysteme, z. B. verwaltet mit Lotus Notes<sup>®</sup> und Microsoft<sup>®</sup> Exchange Server.
- Dokumentverwaltungssysteme, z. B. DB2<sup>®</sup> Datenbanken von Information Integrator for Content.
- Relationale Datenbanken, z. B. IBM DB2, Oracle, Microsoft SQL Server, Microsoft Access und andere Datenbanken, die mit den Standards von Open Database Connectivity (ODBC) kompatibel sind.
- Volltextindizes, z. B. die mit IBM WebSphere<sup>®</sup> Portal, Domino<sup>™</sup> Domain Index, Microsoft Index Server und Microsoft Site Server erstellt.
- Lotus-Repositorys einschließlich Notes-Datenbanken, Domino.Doc<sup>®</sup>-Bibliotheken und -Ablagen (Cabinets), Lotus QuickPlace-Platzierungen und Lotus Discovery Server<sup>™</sup>-Wissenskarten (K-maps).
- Sofortnachrichtensysteme, z. B. Lotus Sametime<sup>®</sup>. Mit dieser Funktion können Sie Abfragen an Personen mit Wissen anstatt lediglich an durchsuchbare Datenrepositorys richten.
- LDAP-Verzeichnisse (LDAP = Lightweight Directory Access Protocol), z. B. verwaltet mit IBM SecureWay<sup>®</sup>, Domino LDAP Server und Exchange LDAP Server.
- Dateisysteme. Sie können Textdateien durchsuchen, die lokal auf Netzlaufwerken gespeichert sind. Komprimierte oder verschlüsselte Dateien können nicht durchsucht werden.

Mit den Extended Search-Anwendungsprogrammierschnittstellen (APIs) C++ und Java<sup>™</sup> können Sie die Unterstützung auf andere Typen von Quellen erweitern, wie z. B. hier nicht erwähnte proprietäre Datenbanken.

## Funktionsweise des Extended Search-Wrappers

Im Modell einer strukturierten relationalen Datenbank werden die Spalten in einem konsistenten Format benannt und dargestellt. Mit Hilfe dieser Funktion können Sie präzise Rechenoperationen ausführen und Daten aus verschiedenen Tabellen durch Vergleichen spezifischer Spaltenwerte miteinander vergleichen. Sie können außerdem auch andere Typen von Analysen ausführen, wie z. B. Objekte in einer Tabelle auflisten, die in einer anderen Tabelle fehlen.

Unstrukturierte Daten dagegen sind oftmals als freier Text gespeichert. Typischerweise sind nur wenige oder überhaupt keine Metadaten vorhanden, mit deren Hilfe Sie Informationen nach Spaltennamen abfragen können. Eine Suche für unstrukturierte Daten bezieht sich mehr auf das Suchen von Daten, die mit den benutzerdefinierten Schlüsselworten übereinstimmen, als auf Rechenkriterien.

Der Extended Search-Wrapper kombiniert diese beiden Suchtechniken. Mit dem Wrapper können Sie strukturierte Abfragesprache zum Suchen von unstrukturiertem Inhalt in einer Extended Search-Domäne verwenden. Danach können Sie analytische oder relationale Operationen für die Suchergebnisse ausführen.

Sie geben Abfragen aus, indem Sie SQL-Anweisungen eingeben, die sich auf eine bestimmte DB2-Tabelle (eine Tabelle mit einem Kurznamen) beziehen. Extended Search führt die Suche entsprechend der SQL-Kriterien aus und füllt die Kurznamentabelle mit den Ergebnisdaten. Da die Suchergebnisse als Tabelle ausgegeben werden, sind die Daten für Operationen mit anderen Datenbanktabellen, einschließlich anderer Kurznamentabellen, verfügbar.

Wenn Sie eine Suchanforderung mit dem Wrapper übergeben, können Sie Daten von einer beliebigen Extended Search-Quelle abrufen, die einer Kurznamentabelle zugeordnet ist. Sie können diese Daten in andere Datenquellen innerhalb Ihres Systems zusammengeschlossener Datenbanken integrieren, ohne die Daten aus der ursprünglichen Datenquelle verschieben zu müssen. Die Suchergebnisse erscheinen als einzelne Ergebnismenge, unabhängig davon, wie viele Quellen Antworten auf die Abfrage zurückgeben.

Die nachfolgende Abbildung zeigt, wie der Extended Search-Wrapper die verschiedenen Datenquellen in einer Extended Search-Domäne mit einem System zusammengeschlossener Datenbanken verbindet. Der Wrapper greift auf Daten eines oder mehrerer ferner Extended Search-Server zu und ruft diese ab. Stellt der Wrapper eine Verbindung zu einem Extended Search-Server her, der mit anderen Extended Search-Servern verbunden ist, können Suchergebnisse von mehreren Servern zurückgegeben werden.

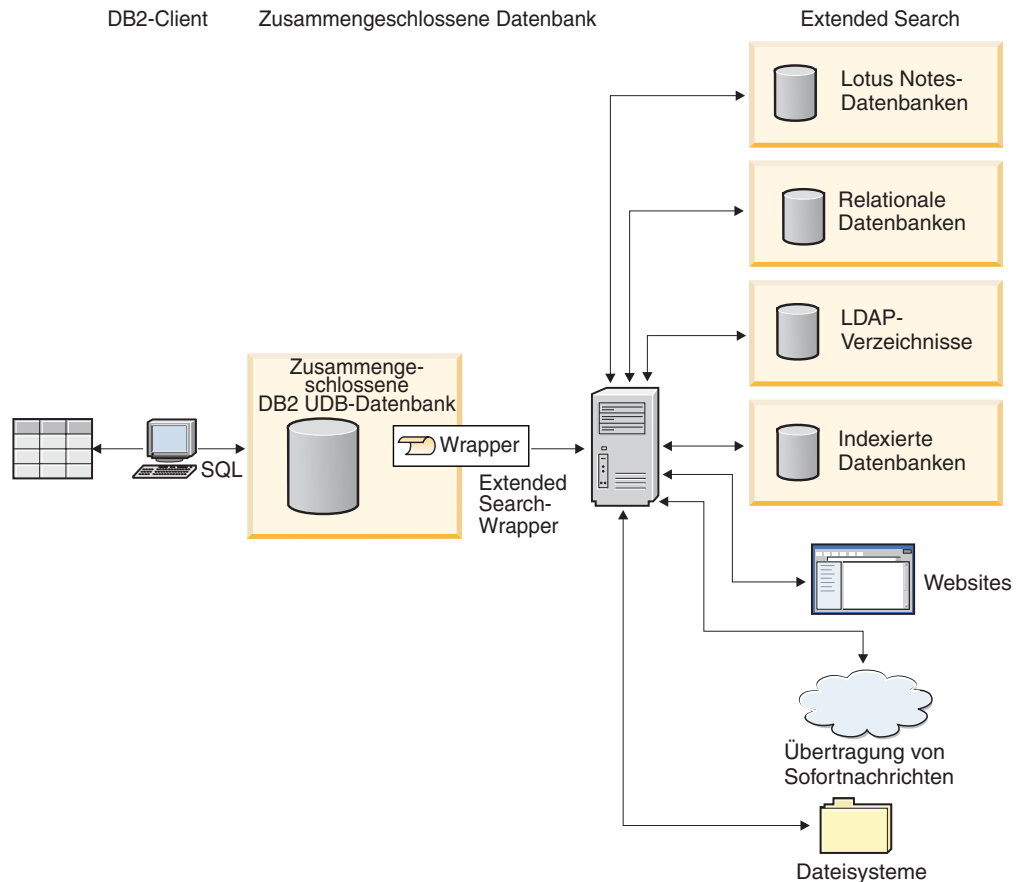


Abbildung 24. Funktionsweise des Extended Search-Wrappers

#### Zugehörige Tasks:

- „Hinzufügen von Extended Search-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 270

## Kurznamen für Extended Search

Beim Datenmodell von Extended Search besteht ein Dokument aus mindestens einem Feld. Eine Gruppe von Dokumenten stellt eine Datenquelle dar. Sie können eine beliebige Anzahl an Datenquellen zu einer Kategorie zusammenfassen und diese Datenquellen dann als Gruppe durchsuchen und verwalten.

Um zu gewährleisten, dass Benutzer nur auf die Datenquellen zugreifen können, die sie tatsächlich benötigen, muss eine Kategorie mindestens einer Anwendung zugeordnet sein. In diesem Zusammenhang sind Anwendungen als eine Art Gruppierung von Benutzern zu verstehen, um den Zugriff und die Suchfunktionalitäten zu steuern. Eine Mitarbeiteranwendung kann beispielsweise dieselben Datenquellen enthalten wie eine Finanzanwendung. Dennoch ist es nicht unbedingt erforderlich, dass die Benutzer der einzelnen Anwendungen auf dieselben Felder in diesen Datenquellen zugreifen.

Beim Registrieren von Kurznamen identifizieren Sie die Anwendungen, Kategorien, Datenquellen und Datenquellenfelder, die Sie durchsuchen wollen. Diese Entitäten müssen in der Extended Search-Konfigurationsdatenbank vorhanden sein.

Um eine Datenquelle von Extended Search mit dem Extended Search-Wrapper zu durchsuchen, müssen Sie für die Quelle einen Kurznamen erstellen.

Der Inhalt der Kurznamentabelle spiegelt den Status der Extended Search-Konfigurationsdatenbank zu dem Zeitpunkt wider, an dem Sie den Kurznamen registrieren. Wenn ein Extended Search-Administrator die Konfiguration aktualisiert (beispielsweise durch Hinzufügen oder Löschen von Datenquellen oder Feldern), werden diese Änderungen in der Kurznamentabelle nicht widergespiegelt. Wenn eine Kurznamentabelle auf geänderte Daten verweist und Sie den aktuellen Status der Extended Search-Konfigurationsdatenbank verwenden wollen, müssen Sie den Kurznamen ändern oder den Kurznamen löschen und einen neuen Kurznamen erstellen.

Wenn Sie den Kurznamen nicht ändern oder erneut erstellen, empfangen Sie unter Umständen Fehlermeldungen und Suchberichte ohne Ergebnisse, wenn Sie versuchen, Einträge zu durchsuchen, die in der Extended Search-Domäne nicht mehr vorhanden sind.

Zwar kann eine einzelne Kurznamentabelle Informationen zu allen in Extended Search konfigurierten Quellen enthalten. Unter Umständen ist das Erstellen mehrerer Kurznamentabellen jedoch von größerem Nutzen. Um das volle Leistungsspektrum von DB2<sup>®</sup> verwenden zu können, müssen Sie für jeden Datenquellentyp, der mit dem Extended Search-Wrapper durchsucht werden soll, einen eigenen Kurznamen erstellen.

So können Sie beispielsweise jeweils einen eigenen Kurznamen für Webquellen, für Notes-Datenbanken und für Dateisysteme etc. erstellen. Wenn Sie über derart getrennte Kurznamentabellen verfügen, ist es einfacher, Daten zu verknüpfen, die an den Wrapper zurückgegeben werden, verschiedene Datenquellen auf der Basis von Feldwerten in Beziehung zu setzen und die Ergebnisdaten mit anderen Daten in Ihrem System zusammengeschlossener Datenbanken zu integrieren.

**Zugehörige Konzepte:**

- „Vertikale Tabellen für Extended Search“ auf Seite 267

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Extended Search-Datenquellen“ auf Seite 274

**Zugehörige Referenzen:**

- „Extended Search-Wrapper - Beispielabfragen“ auf Seite 278
- „CREATE NICKNAME-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 613

---

## Vertikale Tabellen für Extended Search

Eine Extended Search-Anwendung kann aus mehreren Kategorien bestehen, die wiederum mehrere Datenquellen enthalten können. Da für jede Datenquelle eigene Konventionen für Feldnamen verwendet werden, können Überschneidungen von Feldern zu einer leeren Gruppe führen. Wenn Sie Datenquellenfelder für benutzerdefinierte Spalten in Kurznamentabellen zuordnen und Suchergebnisse als horizontale Tabelle darstellen, enthält die Tabelle unter Umständen eine Anzahl von Spalten, die nicht verarbeitet werden kann. Wenn viele Zeilen lediglich einige wenige Spalten mit Daten enthalten, sieht die Tabelle weitgehend leer aus. Beispiel:



Column_1	Column_2	Column_3
Value_11		
	Value_22	
Value_31		Value_33

In Extended Search können Sie die Darstellung von Ergebnissen steuern, indem Sie zugeordnete Felder definieren. Mit Hilfe zugeordneter Felder können Sie Inhalte kombinieren, die einen gemeinsamen Zweck erfüllen, in verschiedenen Quellen jedoch unterschiedliche Namen haben. Sie können beispielsweise ein zugeordnetes Feld namens 'EmployeeNumber' (Mitarbeiternummer) erstellen, um Ergebnisdaten aus Feldern darzustellen, die in verschiedenen Quellen den Namen 'EmpNum', 'EmpNo' und 'EmpID' haben. Ohne diese Zuordnungsfunktion müssten Sie eine Kurznamenspalte für jeden einzelnen Feldnamen definieren, anstelle einer einzigen Spalte für das zugeordnete Feld.

Die Zuordnung von Feldern ist dann nützlich, wenn Sie die Namen der Felder kennen, die Sie miteinander in Beziehung setzen wollen. Bei einigen Anwendungen müssen jedoch sehr viele Felder aus vielen verschiedenen Datenquellen miteinander in Beziehung gesetzt werden. Die Abhängigkeiten zwischen den Feldern, insbesondere für unstrukturierte Daten, sind möglicherweise nicht im Voraus bekannt. Dies macht es schwierig, sinnvolle Kurznamentabellen zu definieren und zu strukturieren. Um diesen Anwendungstyp zu unterstützen, können Sie mit dem Extended Search-Wrapper eine vertikale Kurznamentabelle erstellen.

Wenn Sie eine Kurznamentabelle für Extended Search erstellen, können Sie die Option VERTICAL\_TABLE aktivieren. Diese Option gibt alle Felder zurück, die gemäß der Definition der Extended Search-Konfigurationsdatenbank für die Rückgabe in einer Datenquelle konfiguriert sind. Verwenden Sie diese Option, wenn Sie nicht sicher sind, welche Spalten in Ihrer Suche relevant sind, oder welche Spalten relevant sind, wenn Sie Abfragen für die Nachbearbeitung durchführen oder Ergebnismengen verknüpfen wollen.

Jede Zeile in der vertikalen Tabelle enthält Informationen zu einem Feld, das in der Ergebnismenge zurückgegeben wurde. Für jede Zeile gibt Extended Search den Namen der Quelle zurück, aus der das Feld stammt, sowie den Feldnamen, dessen Wert und Datentyp (Datum, ganze Zahl etc.). Während die Ergebnisse in einer horizontalen Tabelle auf mehrere Spalten verstreut sind, enthält die vertikale Tabelle viele Zeilen mit Daten und ist daher dicht gefüllt. Beispiel:

Field_Name	Field_Value	Field_Datatype
Column_1	Value_11	VARCHAR
Column_2	Value_22	DATE
Column_1	Value_31	VARCHAR
Column_3	Value_33	VARCHAR

Beim Abfragen der Tabelle können Sie für diese Daten SQL-Operationen durchführen, und es ist möglich, alle Spaltenkennsätze abzufragen. Beispiel:

```
Field_Value LIKE '%IBM%'
```

Da die Option VERTICAL\_TABLE Informationen zu allen Feldern der Datenquelle zurückgibt, die zurückgegeben werden können, müssen bestimmte benutzerdefinierte Spalten unter Umständen nicht abgefragt werden. Wenn Sie diese Option

aktivieren und anschließend die Anweisung SELECT absetzen, um benutzerdefinierte Spalten zu durchsuchen, kommen in den Suchergebnissen einige Informationen möglicherweise doppelt vor. Wenn Sie benutzerdefinierte Spalten definieren, können sie diese Spalten jedoch in Verknüpfungen mit anderen Tabellen in Ihrem System zusammengeschlossener Datenbanken verwenden.

Die folgenden beiden Tabellen zeigen eine Übersicht der vom System bereitgestellten Spalten, die Extended Search für jede Zeile in einer vertikalen Kurznamentabelle zurückgibt.

Der Wrapper gibt für jeden Kurznamen stets die folgenden drei fixierten Spalten zurück.

Spaltenname	Datentyp	Beschreibung
DOC_ID	VARCHAR(512)	Die Dokumentkennung, die für jeden Eintrag in einer Suchergebnismenge eindeutig ist.
DOC_RANK	INTEGER	Die Rangordnung des Dokuments nach Relevanz.
CLIENT_LOCALE	VARCHAR(5)	Die Ländereinstellung des Clients für die Suchanforderung. Wenn die SQL-Abfrage die Ländereinstellung des Clients nicht zur Verfügung stellt, wird 'enUS' als Standardeinstellung verwendet.

Die folgenden fixierten Spalten werden vom Wrapper nur dann erstellt, wenn die Option VERTICAL\_TABLE aktiviert wurde.

Spaltenname	Datentyp	Beschreibung
DATASOURCE_NAME	VARCHAR(128)	Der Name der Datenquelle, aus der das Suchergebnis stammt.
FIELD_NAME	VARCHAR(128)	Der Name eines Feldes, das mit dem Suchergebnis zurückgegeben wurde.
FIELD_VALUE	VARCHAR(4096)	Der Wert eines Feldes, das in einer Ergebnismenge zurückgegeben wurde. Ist der Feldwert länger als die maximal zulässige Länge der Kurznamenspalte (der Wert für VARCHAR), wird der Feldwert abgeschnitten. Der Token ES_TRUNCATE am Ende der Spalte gibt an, dass der Wert unvollständig ist.
FIELD_DATATYPE	SMALLINT	Eine ganze Zahl, die den Datentyp des Feldwerts angibt.
		384 DATE 448 VARCHAR 496 INTEGER

Vertikale Tabellen, in denen Ergebnisdaten als VARCHAR-Werte gespeichert werden, lassen sich möglicherweise nur schwierig abfragen. Um eine genauere Suche durchzuführen, müssen Sie in der Extended Search-Konfigurationstabelle zugeordnete Felder erstellen und diese anschließend in der Kurznamentabelle definieren. Mit Hilfe zugeordneter Felder können Sie eine übersichtliche horizontale Tabelle für Suchergebnisse erstellen. Außerdem wird hierdurch die Möglichkeit optimiert,

relationale Operationen für die Ergebnisse durchzuführen, und diese in Abfragen zu kombinieren, in die andere Tabellen in Ihrem System zusammengeschlossener Datenbanken einbezogen sind.

Weitere Informationen zum Definieren zugeordneter Felder in Extended Search finden Sie unter *Extended Search Administration* auf der Seite 'Resources' der folgenden Website von IBM® Lotus® Extended Search:

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

**Zugehörige Konzepte:**

- „Kurznamen für Extended Search“ auf Seite 266

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Extended Search-Datenquellen“ auf Seite 274

**Zugehörige Referenzen:**

- „Extended Search-Wrapper - Beispielabfragen“ auf Seite 278
- „CREATE NICKNAME-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 613

---

## Hinzufügen von Extended Search zu einem Server mit zusammengeschlossenen Datenbanken

### Hinzufügen von Extended Search-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Extended Search-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken auf Extended Search-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

**Voraussetzungen:**

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.
- Stellen Sie vor Verwendung des Extended Search-Wrappers sicher, dass die zu durchsuchenden Quellen in der Extended Search-Konfigurationsdatenbank eingerichtet sind. Übergeben Sie einige Abfragen über den Extended Search-Client, um zu überprüfen, ob die Quellen durchsucht werden können, bevor Sie mit der Suche über den Extended Search-Wrapper beginnen.

**Vorgehensweise:**

Gehen Sie wie folgt vor, um Extended Search-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Registrieren Sie den Wrapper.
2. Registrieren Sie die Serverdefinition.
3. Optional: Erstellen Sie die Benutzerzuordnungen.
4. Registrieren Sie Kurznamen für die Extended Search-Datenquellen.
5. Optional: Registrieren Sie angepasste Funktionen für den Extended Search-Wrapper.

#### **Zugehörige Konzepte:**

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

#### **Zugehörige Tasks:**

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Registrieren des Extended Search-Wrappers“ auf Seite 271
- „Registrieren des Servers für Extended Search-Datenquellen“ auf Seite 272
- „Registrieren von Kurznamen für Extended Search-Datenquellen“ auf Seite 274
- „Registrieren von Benutzerzuordnungen für Extended Search-Datenquellen“ auf Seite 273
- „Registrieren der angepassten Funktionen für den Extended Search-Wrapper“ auf Seite 275

#### **Zugehörige Referenzen:**

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## **Registrieren des Extended Search-Wrappers**

Die Registrierung des Extended Search-Wrappers ist Teil der umfangreicheren Task des Hinzufügens von Extended Search-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf Extended Search-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

#### **Vorgehensweise:**

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Bibliotheksdatei des Wrappers aus.

Um beispielsweise einen Wrapper mit dem Namen NotesDBwrapper auf dem Server mit zusammengeschlossenen Datenbanken unter Windows zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER NotesDBwrapper LIBRARY 'db2uies.dll'
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen

zur Angabe des korrekten Namens in der Anweisung CREATE WRAPPER finden Sie in der Liste der Bibliotheksdateien von Extended Search-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den Extended Search-Wrapper.

**Zugehörige Tasks:**

- „Registrieren des Servers für Extended Search-Datenquellen“ auf Seite 272

**Zugehörige Referenzen:**

- „Bibliotheksdateien für den Extended Search-Wrapper“ auf Seite 272
- „CREATE WRAPPER-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 611

## Bibliotheksdateien für den Extended Search-Wrapper

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den Extended Search-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, werden 3 Bibliotheksdateien zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden folgende Wrapperbibliotheksdateien zum Verzeichnispfad hinzugefügt: libdb2uies.a, libdb2uiesF.a und libdb2uiesU.a.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 66. Positionen und Dateinamen von Extended Search-Wrapperbibliotheken*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2uies.a
Linux	/opt/IBM/db2/V8.1/lib	libdb2uies.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2uies.so
Windows	%DB2PATH%\bin	db2uies.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist C:\Program Files\IBM\SQLLIB.

**Zugehörige Tasks:**

- „Registrieren des Extended Search-Wrappers“ auf Seite 271

## Registrieren des Servers für Extended Search-Datenquellen

Diese Task ist Teil der Haupttask für das Hinzufügen von Extended Search-Datenquellen zu einem System zusammengeschlossener Datenbanken. Nachdem Sie den Wrapper registriert haben, müssen Sie eine entsprechende Serverdefinition zum Identifizieren des fernen Extended Search-Servers erstellen, den Sie in Ihr System zusammengeschlossener Datenbanken integrieren möchten. Mit Hilfe dieser Definition kann der Wrapper eine Verbindung zum Extended Search-Server herstellen.

### Vorgehensweise:

Geben Sie zum Registrieren des Extended Search-Servers eine Anweisung CREATE SERVER über den DB2-Befehlszeilenprozessor aus.

Geben Sie beispielsweise folgende Anweisung aus, um einen Server mit dem Namen es1 für einen Wrapper namens myESwrapper zu registrieren. Der Extended Search-Server verwendet den Standardwert für den Port.

```
CREATE SERVER es1 WRAPPER myESwrapper OPTIONS (ES_HOST 'my.server.com')
```

Geben Sie folgende Anweisung aus, um diesen Server zu erstellen und dabei die Tracefunktion für alle Nachrichtenebenen (kritisch, nicht kritisch, Warnung und Information) zu aktivieren, wobei die Tracenachrichten in eine Datei mit dem Namen es1wrapper.log im Verzeichnis wrapper geschrieben werden:

```
CREATE SERVER es1 WRAPPER myESwrapper OPTIONS (ES_HOST 'my.server.com',  
ES_TRACING 'ON', ES_TRACELEVEL 'CNWI',  
ESTRACEFILENAME '/wrapper/es1wrapper.log')
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für Extended Search-Datenquellen.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für Extended Search-Datenquellen“ auf Seite 274

### Zugehörige Referenzen:

- „CREATE SERVER-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 611

## Registrieren von Benutzerzuordnungen für Extended Search-Datenquellen

Diese Task ist ein optionaler Arbeitsschritt in der Haupttask für das Hinzufügen von Extended Search-Datenquellen zu einem System zusammengeschlossener Datenbanken.

Benutzerzuordnungen bieten eine Möglichkeit für die Authentifizierung des Zugriffs von Benutzern, die eine Extended Search-Datenquelle mit dem Extended Search-Wrapper abfragen. Übergibt ein Benutzer eine SQL-Abfrage an einen registrierten Extended Search-Kurznamen, ohne dass für diesen Benutzer Benutzerzuordnungen definiert sind, so verwendet der Extended Search-Wrapper eine Standard-Benutzer-ID mit Kennwort und versucht, die Daten von dem fernen Extended Search-Server abzurufen. Erfordert eine abgefragte Datenquelle Authentifizierung, wird unter Umständen eine leere Ergebnismenge zurückgegeben.

Um sicherzustellen, dass die korrekte Benutzer-ID mit Kennwort an den Extended Search Server übergeben wird, erstellen Sie in Ihrem System zusammengeschlossener Datenbanken Benutzerzuordnungen für Benutzer, die für die Suche in Extended Search-Quellen berechtigt sind. Wenn Sie eine Benutzerzuordnung erstellen, wird das Kennwort in verschlüsseltem Format in einer DB2-Katalogtabelle gespeichert. Das Kennwort wird auch in gesichertem Format von DB2 über Extended Search an die zu durchsuchenden Quellen übergeben.

Die Sicherheitseinstellungen in der Extended Search-Konfigurationsdatenbank ermitteln, ob die Benutzer-ID und das Kennwort für den Zugriff auf die zu durchsuchenden Quellen berechtigt sind und ob eine zusätzliche Zuordnung der Benutzer-ID ausgeführt wird.

### Vorgehensweise:

Geben Sie zum Registrieren der Extended Search-Benutzerzuordnungen eine Anweisung CREATE USER MAPPING über den DB2-Befehlszeilenprozessor aus. Die Anweisung muss die zuzuordnende DB2-Benutzer-ID und den Extended Search-Server mit den Zieldatenquellen sowie die Benutzer-ID und das Kennwort, die der Benutzer für den Zugriff auf diese Datenquellen benötigt, angeben.

Im folgenden Beispiel wird z. B. die Benutzer-ID des Benutzers user1 so registriert, dass user1 mit dem Extended Search-Server es1 ferne Datenbanken durchsuchen kann.

```
CREATE USER MAPPING FOR user1 SERVER es1 OPTIONS
  (REMOTE_AUTHID 'ESUserId', REMOTE_PASSWORD 'abc123def')
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der Schablone für angepasste Extended Search-Funktionen.

### Zugehörige Tasks:

- „Registrieren der angepassten Funktionen für den Extended Search-Wrapper“ auf Seite 275

### Zugehörige Referenzen:

- „CREATE USER MAPPING-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 613

## Registrieren von Kurznamen für Extended Search-Datenquellen

Diese Task ist Teil der Haupttask für das Hinzufügen von Extended Search-Datenquellen zu einem System zusammengeschlossener Datenbanken. Nach der Registrierung eines Servers muss mindestens ein Kurzname registriert werden. Eine Kurznamentabelle ist eine virtuelle DB2-Tabelle, die eine oder mehrere Quellen in einer Extended Search-Domäne identifiziert. Beim Übergeben einer Abfrage geben Sie den Kurznamen der Quellen an, die Sie durchsuchen möchten.

### Voraussetzungen:

Stellen Sie sicher, dass der Extended Search-Server für die zu erstellenden Kurznamen aktiv ist. Wenn Sie einen Kurznamen erstellen, prüft das System, ob diese Informationen über die zu durchsuchenden Quellen und Felder in der Extended Search-Konfigurationsdatenbank vorhanden sind.

### Vorgehensweise:

Geben Sie zum Registrieren des Extended Search-Kurznamens eine Anweisung CREATE NICKNAME über den DB2-Befehlszeilenprozessor aus.

Geben Sie beispielsweise die folgende Anweisung zur Erstellung einer Kurznamentabelle zum Durchsuchen aller Datenquellen aus, die zur Kategorie 'Web' der Anwendung 'Demo' gehören, die sich auf dem Extended Search-Server 'es1' befindet. Bestätigen Sie die Felder WebTitle und WebDescription, und verwenden Sie die Standardoptionen für die Verarbeitung.

```
CREATE NICKNAME allweb (WebTitle VARCHAR(255), WebDescription VARCHAR(1000))
  FOR SERVER es1 OPTIONS(APPLICATIONID 'Demo', CATEGORY 'Web')
```



Geben Sie folgende Anweisung aus, um eine Kurznamentabelle zum Durchsuchen mehrerer Datenquellen der Anwendung 'Science' zu erstellen. Die Suchergebnisse werden als vertikale Liste mit Spaltennamen dargestellt, der Wert für die Zeitlimitüberschreitung ist auf 60 Sekunden gesetzt, für jede Quelle ist die Rückgabe von bis zu 100 Ergebnisdokumenten zulässig, die Größe der Ergebnismenge wird auf 1000 Einträge erweitert, und die Ergebnisse werden nach Autorennamen sortiert.

```
CREATE NICKNAME stars (Title VARCHAR(80), Author VARCHAR(40),  
    Abstract VARCHAR(200))  
FOR SERVER es1 OPTIONS (APPLICATIONID 'Science',  
    DATASOURCES 'Astronomy;NASA Library;Astrophysics', VERTICAL_TABLE 'yes',  
    TIMEOUT '60', MAXHITS '100', TOTALMAXHITS '1000', SORTFIELD 'Author')
```

Die nächste Task in dieser Tasksequenz ist das Registrieren von Benutzerzuordnungen für den Extended Search-Wrapper.

#### **Zugehörige Konzepte:**

- „Kurznamen für Extended Search“ auf Seite 266
- „Vertikale Tabellen für Extended Search“ auf Seite 267

#### **Zugehörige Tasks:**

- „Hinzufügen von Extended Search-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 270
- „Registrieren von Benutzerzuordnungen für Extended Search-Datenquellen“ auf Seite 273
- „Angaben von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75

#### **Zugehörige Referenzen:**

- „Extended Search-Wrapper - Beispielabfragen“ auf Seite 278
- „CREATE NICKNAME-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 613

## **Registrieren der angepassten Funktionen für den Extended Search-Wrapper**

Diese Task ist ein optionaler Arbeitsschritt in der Haupttask für das Hinzufügen von Extended Search-Datenquellen zu einem System zusammengeschlossener Datenbanken.

Angepasste Funktionen enthalten keinen ausführbaren Code. Nachdem Sie eine Funktion registriert haben, können Sie in Abfragen darauf verweisen, um das Verhalten einer standardmäßigen Suche zu ändern. Mit Hilfe der angepassten Funktion für den Extended Search-Wrapper können Sie präzise Suchausdrücke und Suchinhalte angeben, die nicht als Spalte in der Kurznamentabelle definiert sind.

#### **Einschränkungen:**

- Die Extended Search-Funktion kann nur mit einer WHERE-Klausel aufgerufen werden.
- Die WHERE-Klausel muss mindestens ein Vergleichselement für die Suche enthalten, entweder die Funktion Extended Search oder ein Vergleichselement des Typs "column-name operator constant" (Operatorkonstante für Spaltenname).



- Die Extended Search-Funktion ist eine Schablone für eine Skalarfunktion. Sie muss den Operator EQUAL (=) verwenden, und der Vergleichswert muss eins (1) sein.
- Der erste Parameter in der Extended Search-Funktion dient als Ankerwert für die Identifizierung des Kurznamens, auf den die Funktion angewendet werden soll, z. B. der Rang des Dokuments (DOC\_RANK) innerhalb der Suchergebnisse. Für diesen Parameter müssen Sie ein Feld für ganze Zahlen angeben. Dieser Parameter, der nicht ausgewertet wird, ist besonders dann wichtig, wenn die SQL- Abfrage mehr als einen Kurznamen oder eine Kombination aus Kurznamen und Tabellen enthält. Beispiel:

```
SELECT * FROM es_nickname1, es_nickname2
WHERE eswrapper.es_search(es_nickname1.DOC_RANK, '"IBM"') = 1 AND
      eswrapper.es_search(es_nickname2.DOC_RANK, '"IBM"') = 1
```

#### Vorgehensweise:

Geben Sie zum Registrieren der angepassten Extended Search-Funktion folgende Anweisung CREATE FUNCTION aus:

```
CREATE FUNCTION eswrapper.es_search(integer, varchar(1024))
  RETURNS INTEGER AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION;
```

#### Zugehörige Referenzen:

- „Extended Search-Wrapper - Beispielabfragen“ auf Seite 278
- „Extended Search-Wrapper - Abfragesprache GQL (Generalized Query Language)“ auf Seite 281
- „CREATE FUNCTION-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 616
- „Extended Search-Wrapper - Richtlinien für Abfragen“ auf Seite 276

---

## Abfragen von Extended Search-Datenquellen

### Extended Search-Wrapper - Richtlinien für Abfragen

Abfragen, die mit dem Extended Search-Wrapper durchgeführt werden, müssen in einem bestimmten Format vorliegen. Abfragen, die diese Sprachkriterien nicht genau erfüllen, werden nicht unterstützt. Der vorliegende Abschnitt beschreibt die Richtlinien zum Erstellen von Abfragen und enthält Beispiele zur richtigen und falschen Abfragesyntax.

#### Abfragen von Webquellen in mehreren Sprachen

Die Software von Fremdanbietern, die Extended Search zum Herstellen von Verbindungen zu Webquellen verwendet, unterstützt Sprachen, die die Codepage ISO-8859-1 verwenden (wie beispielsweise Englisch, Französisch, Deutsch, Portugiesisch und Schwedisch). Daher können Sie keine Webquellen durchsuchen, die in Sprachen geschrieben sind, die einen Doppelbyte-Zeichensatz verwenden (wie beispielsweise Koreanisch), bzw. die in bidirektionalen Sprachen wie beispielsweise Hebräisch geschrieben sind oder in anderen Sprachen, die nicht den Zeichensatz ISO-8859-1 verwenden. Der Parser, der die Suchergebnisse verarbeitet, schlägt fehl, wenn er Zeichencodes ermittelt, die für ihn ungültig sind.

## Angeben des Wertes für CLIENT\_LOCALE

Wenn Sie die Spalte CLIENT\_LOCALE in eine WHERE-Klausel einschließen, um den Wert für die Ländereinstellung des Clients anzugeben, müssen Sie zum Angeben der Suchbedingungen ein AND-Vergleichselement verwenden. Mit der Spalte CLIENT\_LOCALE kann kein OR-Vergleichselement verwendet werden.

Beispiele — korrekte Syntax

Die folgenden Beispiele zeigen die korrekte Syntax zum Einschließen der Spalte CLIENT\_LOCALE in eine WHERE-Klausel:

```
WHERE CLIENT_LOCALE = 'enUS' AND  
ESWRAPPER.ES_SEARCH(DOC_RANK, 'IBM')=1
```

```
WHERE ESWRAPPER.ES_SEARCH(DOC_RANK, 'IBM')=1  
AND CLIENT_LOCALE = 'enUS'
```

Beispiele — falsche Syntax

Die Syntax in den folgenden Beispielen ist falsch, da hier mit der Spalte CLIENT\_LOCALE ein OR-Vergleichselement verwendet wird:

```
WHERE CLIENT_LOCALE = 'enUS' OR  
ESWRAPPER.ES_SEARCH(DOC_RANK, 'IBM')=1
```

```
WHERE ESWRAPPER.ES_SEARCH(DOC_RANK, 'IBM')=1  
OR CLIENT_LOCALE = 'enUS'
```

## Angeben von Vergleichselementen für fixierte Spalten von Extended Search

Bei SQL-Anweisungen, die einen Extended Search-Kurznamen enthalten, muss in der WHERE-Klausel ein Vergleichselement für den Kurznamen angegeben werden. Ein Vergleichselement für eine fixierte Spalte von Extended Search gilt jedoch nicht als Vergleichselement.

Beispiele — falsche Syntax

Die Abfrage in folgendem Beispiel ist falsch, da sie kein Vergleichselement enthält:

```
SELECT * FROM ES_NICKNAME
```

Die Abfrage in folgendem Beispiel ist ebenfalls falsch, da sich das einzige Vergleichselement auf eine fixierte Spalte bezieht:

```
SELECT * FROM ES_NICKNAME WHERE DOC_RANK < 20
```

## Angeben nicht gebundener Vergleichselemente

Ein Vergleichselement für eine benutzerdefinierte Spalte wird vom Extended Search-Wrapper nur dann verarbeitet, wenn der Vergleichselementwert eine Konstante ist. Ist der Vergleichselementwert nicht gebunden, wird das Vergleichselement von der DB2-Steuerkomponente verarbeitet. Ist das einzige Vergleichselement in einer SQL-Anweisung ein nicht gebundenes Vergleichselement, führt dies zu einem Fehler. Ein Extended Search-Kurzname erfordert ein Vergleichselement, das vom Extended Search-Wrapper verarbeitet werden kann.

## Beispiele — korrekte Syntax

Die WHERE-Anweisung im folgenden Beispiel zeigt ein Vergleichselement, das vom Extended Search-Wrapper verarbeitet wird:

```
SELECT *
FROM   ES_NICKNAME
WHERE  Author = 'Ernest Hemingway'
```

## Beispiele — falsche Syntax

Die WHERE-Anweisung im folgenden Beispiel zeigt ein Vergleichselement, das von DB2 verarbeitet wird:

```
SELECT *
FROM   ES_NICKNAME_1, ES_NICKNAME_2
WHERE  ES_NICKNAME_1.Author = ES_NICKNAME_2.Author
```

## Verknüpfen von Abfragen mit einem OR-Vergleichselement

Unterschiedliche Kurznamentabellen bzw. Kurznamentabellen und Datenbanktabellen, die durch ein einfaches OR-Vergleichselement verknüpft werden, können vom Extended Search-Wrapper nicht durchsucht werden. Ein OR-Vergleichselement kann nur innerhalb desselben Kurznamens verwendet werden.

## Beispiele — falsche Syntax

```
SELECT *
FROM   ES_Nickname as N1, TABLE as T1
WHERE  N1.Column1 = 'abc' OR T1.Column1 = 'abc'
```

```
SELECT *
FROM   ES_Nickname_1 as N1, ES_Nickname_2 as N2
WHERE  N1.UserdefCol = 'abc' OR N2.UserdefCol = 'cdf'
```

```
SELECT *
FROM   ES_Nickname_1 as N1, ES_Nickname_2 as N2
WHERE  ESWRAPPER.ES_SEARCH(N1.DOC_RANK, '"IBM"')=1 OR
      ESWRAPPER.ES_SEARCH(N2.DOC_RANK, '"LOTUS"')=1
```

### Zugehörige Tasks:

- „Registrieren der angepassten Funktionen für den Extended Search-Wrapper“ auf Seite 275

### Zugehörige Referenzen:

- „Extended Search-Wrapper - Beispielabfragen“ auf Seite 278
- „Extended Search-Wrapper - Abfragesprache GQL (Generalized Query Language)“ auf Seite 281

## Extended Search-Wrapper - Beispielabfragen

Um Abfragen mit dem Extended Search-Wrapper auszuführen, geben Sie in Ihren SQL-Anweisungen einen registrierten Kurznamen und Kurznamenspalten in derselben Weise an wie einen typischen DB2-Tabellennamen und Tabellenspalten.

In diesem BeispielsuchszENARIO möchte ein Krankenhausteam die neuesten medizinischen Forschungsergebnisse durchsuchen und vergleichen. Das Krankenhaus verwendet zum Durchsuchen einer großen Anzahl verschiedener Quellen einen Extended Search-Server. Die Extended Search-Domäne beinhaltet eine Anwendung

namens MedResearch mit verschiedenen Kategorien, die für die Suche in dokumentbasierten Datenbanken, Mail-Servern und im Internet konfiguriert sind. Neben der auszuführenden Suche möchte das Team die Ergebnisse verschiedener Suchen miteinander vergleichen. Beispielsweise sollen Personen, die innerhalb eines bestimmten Zeitrahmens Artikel veröffentlicht haben, Personen, die in letzter Zeit Kräuter und Vitamine erworben haben, Personen, die alternative Heilmethoden mit Kollegen per E-Mail diskutiert haben und Personen, die die Erneuerung ihrer medizinischen Lizenz beantragt haben, identifiziert werden. Der Extended Search-Wrapper bietet die Lösung für all diese Anforderungen durch die Möglichkeit, unstrukturierte Extended Search-Daten zum strukturierten Abrufen in DB2 zu integrieren.

Das Krankenhausteam beschließt die Erstellung von drei Kurznamen, einen zum Durchsuchen von Dokumentrepositorys, einen zum Durchsuchen von E-Mail-Systemen, und einen zum Durchsuchen von spezifischen Internetquellen. Die Felder für Eigner und Datum werden als zugeordnete Felder in der Extended Search-Konfigurationsdatenbank definiert, sodass der Benutzer diese Felder unabhängig von ihrer Benennung in den ursprünglichen Datenquellen als Verknüpfungen verwenden kann.

Dokumentkurzname:

```
CREATE NICKNAME MedDocs ( Owner    VARCHAR(80),
                          Date     DATE,
                          Title    VARCHAR(80),
                          Abstract  VARCHAR(200) )
FOR SERVER esServer OPTIONS ( APPLICATIONID 'MedResearch',
                              CATEGORY 'AMA Library;Medical Records;Pharmacy',
                              VERTICAL_TABLE 'YES',
                              TIMEOUT '60', MAXHITS '100',
                              TOTALMAXHITS '1000' )
```

E-Mail-Kurzname:

```
CREATE NICKNAME MedMail ( Owner    VARCHAR(80),
                           To      VARCHAR(80),
                           Date    DATE,
                           Subject  VARCHAR(80) )
FOR SERVER esServer OPTIONS ( APPLICATIONID 'MedResearch',
                              CATEGORY 'Exchange Server;Lotus Notes',
                              VERTICAL_TABLE 'YES', )
                              TIMEOUT '60', MAXHITS '100',
                              TOTALMAXHITS '1000' )
```

Internetkurzname:

```
CREATE NICKNAME MedWeb ( WebTitle VARCHAR(255),
                          WebDescription VARCHAR(1000) )
FOR SERVER esServer OPTIONS ( APPLICATIONID 'MedResearch',
                              DATASOURCES 'Google!;Alta Vista;CNN',
                              TOTALMAXHITS '500' )
```

Die folgende Abfrage sucht nach Dokumenten, die den Ausdruck 'Artificial Liver' im Titel und die Abkürzung 'MARS' im Dokumentinhalt enthalten. Die Ergebnismenge soll Dokumente ausschließen, die vor dem Jahr 2001 veröffentlicht wurden.

```
SELECT OWNER, DOC_CONTENT
FROM   MedDocs
WHERE  ESWRAPPER.ES_Search(DOC_RANK, '( ( TOKEN:EXACT "MARS") AND
                                     ( "TITLE" IN "Artificial Liver") AND
                                     ("DATE" >= "01/01/2001" ) ) ') = 1
```

Die folgende Abfrage sucht nach E-Mails, die während der letzten paar Monate verfasst wurden und alternative Heilmethoden diskutieren:

```
SELECT *
FROM MedMail
WHERE ESWRAPPER.ES_Search(DOC_RANK, '(
                                ("SUBJECT" IN "alternative medicine") AND
                                ("DATE" BETWEENI "03/01/2002" AND
                                "09/30/2002") ) ') = 1
```

Die folgende Abfrage durchsucht Internetquellen, die sich auf komplementäre und alternative Medizin (complementary and alternative medicine, CAM) und deren Akzeptanz bei der amerikanischen Bevölkerung bezieht:

```
SELECT WebTitle, WebDescription
FROM MedWeb
WHERE ESWRAPPER.ES_Search(DOC_RANK, '(
                                TOKEN:EXACT "CAM therapy" ) AND
                                ( TOKEN:FUZZY "United States" ) ') = 1
```

Folgende Abfrage sucht nach kürzlich lizenzierten Ärzten, die große Mengen an Kräutern oder Vitaminen von der Krankenhausapotheke erworben haben. Die Abfrage gleicht dann die Namen von Ärzten mit Personen ab, die E-Mails über alternative Heilmethoden verfasst haben.

```
SELECT N2.OWNER, N2.DATE
FROM MedDocs as N1,
MedMail as N2
WHERE ESWRAPPER.ES_SEARCH(N1.DOC_RANK, '(
                                ("LICENSE_DATE" >= "01/01/2002") AND
                                ( ( ( "PRODUCT" = "HERB") OR ("PRODUCT" = "VITAMIN") ) AND
                                ("QUANTITY" > "1000") ) ) ') = 1
AND ESWRAPPER.ES_SEARCH(N2.DOC_RANK, '( "SUBJECT" IN
                                "alternative medicine" ) ') = 1
AND N1.OWNER = N2.OWNER
```

#### **Zugehörige Konzepte:**

- „Kurznamen für Extended Search“ auf Seite 266
- „Vertikale Tabellen für Extended Search“ auf Seite 267

#### **Zugehörige Tasks:**

- „Registrieren von Kurznamen für Extended Search-Datenquellen“ auf Seite 274
- „Registrieren der angepassten Funktionen für den Extended Search-Wrapper“ auf Seite 275

#### **Zugehörige Referenzen:**

- „Extended Search-Wrapper - Abfragesprache GQL (Generalized Query Language)“ auf Seite 281
- „CREATE FUNCTION-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 616
- „CREATE NICKNAME-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 613
- „Extended Search-Wrapper - Richtlinien für Abfragen“ auf Seite 276

## Extended Search-Wrapper - Abfragesprache GQL (Generalized Query Language)

Abfragen, die über den Extended Search-Wrapper an einen Extended Search-Server übergeben werden, können Suchausdrücke in GQL (Generalized Query Language), der Abfragesprache von Extended Search, enthalten.

Nehmen wir beispielsweise an, ein Benutzer möchte in einer relationalen Datenbank, die eine Tabelle mit Informationen über Angestellte nach allen Angestellten suchen, deren Namen mit JO beginnen. Sie könnten folgende Abfrage in GQL ausgeben:

```
(LIKE "EMPLOYEE_NAME" "JO")
```

Sie könnten dieselbe Abfrage in SQL wie folgt ausgeben:

```
SELECT * FROM EMP.TABLE WHERE EMPLOYEE_NAME LIKE JO%
```

Ebenso wie SQL unterstützt der Wrapper die eingebettete Schreibweise, eine Syntax, bei der Operatoren zwischen dem Feldnamen und dem Vergleichswert stehen müssen. Die native Extended Search-GQL-Grammatik verwendet die Präfixnotation, eine Syntax, bei der Operatoren vor den auszuwertenden Feldern und Werten stehen müssen.

Vergleichen Sie folgende Abfrageausdrücke, die nach Dokumenten suchen, die das Wort 'IBM' im Feld 'TITLE' enthalten:

### GQL mit eingebetteter Schreibweise

```
("TITLE" IN "IBM")
```

### GQL mit Präfixnotation

```
(IN "TITLE" "IBM")
```

Wenn Sie eine Abfrage mit dem Extended Search-Wrapper übergeben, konvertiert die API die SQL-Anweisungen mit eingebetteter Schreibweise in GQL-Anweisungen mit Präfixnotation für die Verarbeitung in Extended Search.

Die folgende Syntaxbeschreibung zeigt die Angabe der Backus-Naur-Form für die Extended Search-Grammatik, die Sie in Abfragen verwenden können.

```
expr:          pattern_expr
              | bool_expr
              | field_expr
              | prox_expr

pattern_expr:  STRING
              | token_expr

token_expr:   ( TOKEN [:CASE] [:STEM] [:EXACT] [:WEIGHT "x"]
              [:WILD] [:FUZZY] STRING )

bool_expr:    (expr_list bool_operator [:WEIGHT "x"] expr )

bool_text_expr: (text_expr_list bool_operator [:WEIGHT "x"] text_expr )

text_expr:    pattern_expr
              | bool_text_expr
              | prox_expr

text_expr_list: text_expr
              | text_expr_list text_expr
```

```

expr_list:      expr
               | expr_list expr

field_expr:    ( field_name operator_1 [:WEIGHT "x"] text_expr )
               | ( field_name operator_2 [:WEIGHT "x"] value )
               | ( field_name operator_3 [:WEIGHT "x"] value_1 AND value_2 )
               | ( field_name operator_4 value )

prox_expr:     ( prox_op [:COUNT "x"][:ORDER][:MATH "y"][:WEIGHT "x"]
               expr_list expr )

prox_op:       DOCUMENT
               | PARAGRAPH
               | SENTENCE
               | WORD
               | CHARACTER

operator1:     START
               | END
               | IN
               | =

operator_2:    =
               | >
               | >=
               | <
               | <=
               | EQ
               | GT
               | GTE
               | LT
               | LTE

operator_3:    BETWEENI
               | BETWEENE
               | LIKE

bool_operator: AND
               | OR
               | NOT

```

Vollständige Informationen über die GQL-Grammatik finden Sie unter *Extended Search Programming* auf der Ressourcenseite der IBM Lotus Extended Search-Website:

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

#### Zugehörige Tasks:

- „Registrieren der angepassten Funktionen für den Extended Search-Wrapper“ auf Seite 275

#### Zugehörige Referenzen:

- „Extended Search-Wrapper - Beispielabfragen“ auf Seite 278
- „CREATE FUNCTION-Anweisungssyntax - Extended Search-Wrapper“ auf Seite 616
- „Extended Search-Wrapper - Richtlinien für Abfragen“ auf Seite 276

---

## Nachrichten für den Extended Search-Wrapper

In diesem Abschnitt werden Nachrichten beschrieben, die beim Arbeiten mit dem Extended Search-Wrapper auftreten können.

Tabelle 67. Vom Extended Search-Wrapper ausgegebene Nachrichten

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug auf Grund eines nicht schwer wiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: Interner Fehler des Extended Search-Wrappers - Ursachencode: xxx.)	Zeichnen Sie den Ursachencode auf (eine Zahl zwischen 901 und 999) und wenden Sie sich an die IBM Softwareunterstützung.
SQL0973N	Nicht genügend verfügbarer Speicherkapazität im Zwischenspeicher der Anwendung, um die Anweisung zu verarbeiten.	Der Extended Search-Wrapper konnte dem Zwischenspeicher der Anwendung nicht genügend Speicherkapazität zuordnen. Um das Problem zu beheben, erweitern Sie die Größe des Zwischenspeichers der Anwendung und versuchen Sie, die Anweisung erneut auszugeben. Beispiel: <pre>db2 update db cfg for db-name using heap-name heap-size</pre> <p>Tritt der Fehler nach der Erhöhung dieses Wertes weiterhin auf, wenden Sie sich an die IBM Softwareunterstützung.</p>
SQL1822N	Es wurde ein unerwarteter Fehlercode "<fehlercode>" von der Datenquelle "Extended Search-Wrapper" empfangen. Zugeordneter Text und Token sind "<token>".	Der ferne Extended Search-Server gab während der Verarbeitung der Suchanforderung einen Fehler zurück. Der Fehler gab außerdem ein Token zurück, das die Fehlerursache auf dem fernen Server angibt. Falls die Tracefunktion für den Extended Search-Server aktiviert ist, überprüfen Sie die Traceprotokolldatei auf diagnostische Hilfe.
SQL1823N	Für den Datentyp "<datentyp>" gibt es vom Server "<servername>" keine Datentypzuordnung.	Eine Spalte in einer Anweisung CREATE NICKNAME oder ALTER NICKNAME verwendet einen Datentyp, der vom Extended Search-System nicht unterstützt wird. Dieser Fehler kann auch während der Abfrageverarbeitung auftreten. Um das Problem zu beheben, wenn es während der Abfrageverarbeitung auftritt, löschen Sie die Kurznamentabelle und erstellen Sie einen neuen Kurznamen.
SQL1825N	Diese SQL-Anweisung kann nicht in einer Umgebung mit zusammengeschlossenen Datenbanken ausgeführt werden.	Die aktuelle SQL-Anweisung kann vom Extended Search-Wrapper nicht ausgeführt werden. Um das Problem zu beheben, konsultieren Sie die Dokumentation zum Extended Search-Wrapper, ändern Sie die SQL-Anweisung wie erforderlich, und übergeben Sie die Anforderung erneut.



Tabelle 67. Vom Extended Search-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1833N	Die Verbindung zum fernen Extended Search-Server "<hostname>" an Port "<portnummer>" konnte nicht hergestellt werden oder wurde beendet.	Der Extended Search-Wrapper versuchte, zum fernen Extended Search-Server am angegebenen Port eine Verbindung herzustellen. Die Verbindung konnte jedoch nicht hergestellt werden oder wurde vom fernen Server beendet. Überprüfen Sie den Hostnamen und die Portnummer des fernen Extended Search-Servers, vergewissern Sie sich, dass der Extended Search-Server aktiv ist, und versuchen Sie es erneut.
SQL1834N	Die benutzerdefinierte Spalte "<spaltenname>" ist mit einer festen Spalte für den Wrapper "<wrappername>" identisch, verwendet jedoch einen anderen Datentyp.	Eine Anweisung CREATE NICKNAME oder ALTER NICKNAME enthält eine benutzerdefinierte Spalte, die denselben Namen hat wie eine fixierte Spalte für den angegebenen Extended Search-Wrapper, jedoch einen anderen Datentyp verwendet. Fixierte Spalten müssen in der Spaltendefinition einer Anweisung CREATE NICKNAME nicht angegeben werden. Wenn Sie sie angeben, stellen Sie sicher, dass Name, Datentyp und Datentyp-länge der fixierten Spalte mit der Definition für die fixierte Spalte übereinstimmen. Der Name oder Datentyp einer fixierten Spalte kann nicht geändert werden.
SQL1835N	Das Extended Search-Objekt "<objektname>" des Typs "<objekttyp>" konnte auf dem fernen Extended Search-Server "<hostname>" nicht gefunden werden.	Das angegebene Extended Search-Objekt konnte auf dem angegebenen fernen Extended Search-Server nicht gefunden werden. Überprüfen Sie, ob der Objektname auf diesem Extended Search-Server definiert ist und mit dem angegebenen Objekttyp übereinstimmt. Überprüfen Sie außerdem, ob die Schreibweise dieses Objekts korrekt ist.
SQL1836N	Zwischen der benutzerdefinierten Spalte "<spaltenname>" und einem Feldnamen auf dem fernen Extended Search-Server "<hostname>" ist keine Spaltenzuordnung vorhanden.	Keine der Datenquellen in einer Option DATASOURCE oder CATEGORY enthält einen Feldnamen, der mit dem angegebenen benutzerdefinierten Spaltennamen übereinstimmt. Überprüfen Sie, ob der Spaltenname ein Feld in mindestens einer der Datenquellen in der Option DATASOURCE bzw. in mindestens einer der Datenquellen einer Kategorie in der Option CATEGORY ist, und übergeben Sie die Anweisung erneut.
SQL1837N	Die erforderliche Option "<optionsname>" des Typs "<objekttyp>" für den Wrapper "<wrappername>" kann nicht gelöscht werden.	Sie können eine erforderliche Option nicht löschen. Ändern Sie in der Anweisung ALTER die Verwendung von SET in DROP. Korrigieren Sie die Suchanweisung und übergeben Sie die Anforderung erneut. Informationen über das Erstellen von gültigen SQL-Suchanweisungen finden Sie im Handbuch 'DB2 SQL Reference'. Enthält die Suchanweisung die Funktion ES_SEARCH, so finden Sie Informationen über die Verwendung der Extended Search-Abfragesprache GQL in der Dokumentation zum Extended Search-Wrapper.

Tabelle 67. Vom Extended Search-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1838N	Die Suchanweisung " <code>&lt;optionsname&gt;</code> " ist keine gültige Extended Search-Abfrage.	Der Extended Search-Wrapper versuchte, die angegebene Suchanweisung zu verarbeiten. Die Abfrage schlug fehl, da in der Anweisung nicht die korrekte Abfragesyntax verwendet wurde. Informationen über das Erstellen von gültigen SQL-Suchanweisungen finden Sie im Handbuch <i>DB2 SQL Reference</i> . Enthält die Suchanweisung die Funktion <code>ES_SEARCH</code> , so finden Sie Informationen über die Verwendung der Extended Search-Abfragesprache GQL in der Dokumentation zum Extended Search-Wrapper.
SQL1839N	Mindestens ein Suchparameter ist ungültig.	Der Extended Search-Wrapper versuchte, die angegebenen Suchparameter zu verwenden. Diese sind jedoch für Extended Search ungültig. Konsultieren Sie die Dokumentation zum Extended Search-Wrapper, korrigieren Sie die ungültigen Parameter, und übergeben Sie die Anforderung erneut.
SQL1881N	" <code>&lt;optionsname&gt;</code> " ist keine gültige Option des Typs " <code>&lt;optionstyp&gt;</code> " für " <code>&lt;objektname&gt;</code> ".	Die angegebene Option ist für das angegebene Objekt ungültig (Wrapper, Server, Kurzname, Spalte oder Benutzerzuordnung). Konsultieren Sie die Dokumentation zum Extended Search-Wrapper, entfernen oder ändern Sie die ungültige Option, und übergeben Sie die Anweisung erneut.
SQL1882N	Die Option " <code>&lt;optionsname&gt;</code> " der Art " <code>&lt;optionstyp&gt;</code> " kann nicht auf " <code>&lt;optionswert&gt;</code> " für " <code>&lt;objektname&gt;</code> " eingestellt werden.	Der angegebene Optionswert ist für das angegebene Objekt ungültig (Wrapper, Server, Kurzname, Spalte oder Benutzerzuordnung). Konsultieren Sie die Dokumentation zum Extended Search-Wrapper, ändern Sie den ungültigen Optionswert, und übergeben Sie die Anweisung erneut.
SQL1883N	" <code>&lt;optionsname&gt;</code> " ist eine erforderliche Option des Typs " <code>&lt;optionstyp&gt;</code> " für " <code>&lt;objektname&gt;</code> ".	Eine erforderliche Option für den Extended Search-Wrapper fehlte in der Anweisung zum Erstellen, Ändern oder Initialisieren des angegebenen Objekts (Wrapper, Server, Kurzname oder Benutzerzuordnung). Konsultieren Sie die Dokumentation zum Extended Search-Wrapper, fügen Sie die erforderliche Option hinzu, und übergeben Sie die Anweisung erneut.

Weitere Informationen zu Nachrichten finden Sie im Handbuch zu *DB2 Fehlernachrichten*. Informationen zu diesem Thema finden Sie außerdem in den Extended Search- Produktnachrichten im Handbuch *Extended Search Administration*. Informationen zu empfangenen Fehlern wegen nicht korrekter GQL-Abfragesyntax finden Sie im Handbuch *Extended Search Programming*. Die Extended Search-Dokumentationen sind auf der Ressourcenseite der IBM Lotus Extended Search-Website verfügbar:

<http://www.lotus.com/products/des.nsf/wdocuments/resources>

**Zugehörige Konzepte:**

- „Nachrichten - Einführung“ in *Fehlernachrichten Band 1*

**Zugehörige Referenzen:**

- „SQLSTATE-Nachrichten“ in *Fehlernachrichten Band 2*
- „sql0900“ in *Fehlernachrichten Band 2*
- „sql1800“ in *Fehlernachrichten Band 2*

---

## Kapitel 14. Konfigurieren des Zugriffs auf HMMER-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in HMMER-Datenquellen gespeichert sind. Der Zugriff auf HMMER-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält folgende Informationen:

- Erläuterungen zu HMMER.
- Liste der auszuführenden Tasks.
- Beispiele zu den benötigten SQL-Anweisungen.
- Liste der dem HMMER-Wrapper zugeordneten Fehlermeldungen.

---

### Was ist HMMER?

HMMER ist ein Anwendungspaket zum Durchsuchen von Gensequenzdatenbanken, die statistische Modelle oder Profil-HMMs (HMM = Hidden Markov Model) verwenden. Sie können das HMMER-Anwendungspaket kostenlos unter folgender Adresse herunterladen: <http://hmmer.wustl.edu/>. Das HMMER-Anwendungspaket kann auf einem separaten HMMER-Server oder auf einem Server mit zusammengeschlossenen Datenbanken installiert werden.

Ein HMM ist ein statistisches Modell der primären Strukturübereinstimmung einer Gensequenzfamilie. Ein HMM basiert auf Wahrscheinlichkeitsmodellen. Sie können ein HMM so trainieren, dass es Muster aus Gensequenzen ohne Alignment erkennt, wenn bisher kein zuverlässiges Alignment bekannt ist. Es ist einfacher und weniger aufwendig, ein erfolgreiches HMM zu trainieren und zu verwenden als sorgfältig ein Profil zu erstellen. Mit einem trainierten HMM können Sie auf Bibliotheken von Hunderten von Profil-HMMs zugreifen und diese in großem Stil auf ganze Genom- oder EST-Analysen (EST = Expressed Sequence Tag) anwenden.

PFAM (Protein Families Database of Alignments and HMMs) ist eine Datenbank für Proteindomänenmodelle. Das HMMER-Anwendungspaket ist eng mit der Erstellung und Verwendung der PFAM-Datenbank verbunden.

Das Paket enthält 9 Programme, von denen jedoch nur zwei von DB2<sup>®</sup> Information Integrator und den Programmen 'hmmpfam' und 'hmmsearch' unterstützt werden.

*Tabelle 68. Vom HMMER-Wrapper unterstützte HMMER-Programme*

HMMER-Programm	Beschreibung
hmmpfam	Verwendet eine bestimmte Gensequenz zum Durchsuchen einer HMM-Datenbank und zum Ermitteln der Familie, zu der die Testgensequenz möglicherweise gehört. Berechnet, inwieweit jedes Modell mit einer spezifischen Sequenz und einer Modelldatenbank übereinstimmt. Die Übereinstimmung wird auf der Grundlage der statistischen Signifikanz ausgedrückt.

Tabelle 68. Vom HMMER-Wrapper unterstützte HMMER-Programme (Forts.)

HMMER-Programm	Beschreibung
hmmsearch	Verwendet ein bestimmtes HMM-Profil zum Durchsuchen einer Sequenzdatenbank nach sehr ähnlichen Sequenzübereinstimmungen.

Benutzer oder Anwendungen geben Anweisungen für SQL-Abfragen mit HMMER-spezifischen Vergleichselementen an den Server mit zusammengeschlossenen Datenbanken aus. Die Vergleichselemente in diesen Anweisungen werden den Befehlszeilenoptionen im Programm 'hmmpfam' bzw. 'hmmsearch' zugeordnet.

Der HMMER-Wrapper setzt die Abfrageanweisungen in ein Format um, das vom HMMER-Anwendungspaket interpretiert werden kann, und startet das Programm 'hmmpfam' bzw. 'hmmsearch', um die Abfrage auszuführen.

Ein spezielles Dämonprogramm wird auf dem Server ausgeführt, auf dem das HMMER-Anwendungspaket installiert ist. Dieser Dämon empfängt die Abfrageanforderung vom Server mit zusammengeschlossenen Datenbanken und sendet sie an das HMMER-Anwendungspaket. Das HMMER-Anwendungspaket führt die Abfrage für eine Profildatenbank, z. B. PFAM, aus.

Abb. 25 zeigt die Funktionsweise von HMMER in Ihrem System zusammengeschlossener Datenbanken.

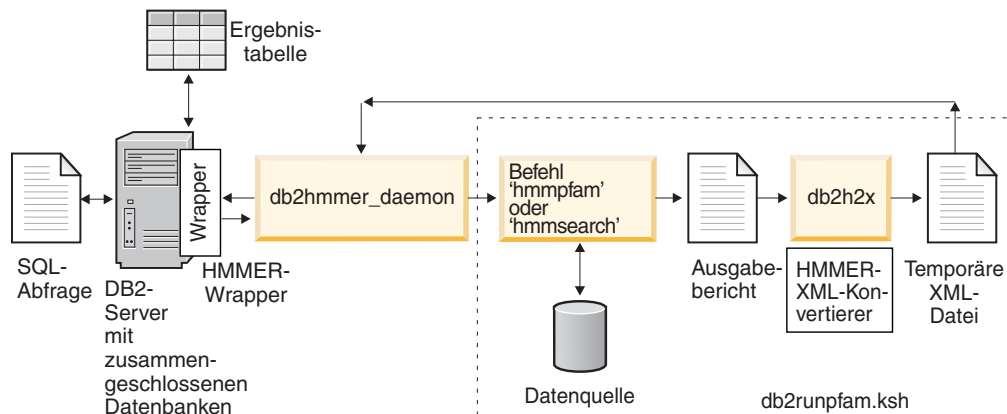


Abbildung 25. Funktionsweise des HMMER-Wrappers

Der Dämon gibt die Ergebnisse an den HMMER-Wrapper zurück. Der Wrapper setzt die Daten in eine relationale Tabelle um und gibt diese Tabelle an den Benutzer bzw. die Anwendung zurück.

Das folgende Beispiel zeigt, wie Informationen aus Profildatenbanken, die von HMMER-Programmen erstellt wurden, extrahiert und als relationale Tabelle angezeigt werden. Das Benutzerhandbuch zu HMMER (<http://hmmmer.wustl.edu/>) enthält Beispiele zum Erstellen von Profildatenbanken sowie ein Lernprogramm für HMMER.

Abb. 26 auf Seite 289 zeigt eine Beispielabfrage anhand der Gensequenz 7LES\_DROME. Die Sequenzen werden in der WHERE-Klausel der Abfrage angegeben.

```

SELECT Model, ModelScore, DomainNumber, DomainScore
FROM myhmms
WHERE HmmQSeq = 'MTMFWQQNVDHQSDSEQDKAKGAAPTAKRLNISFNVKIHAVNVNTKMTTTH
INQQAPGTSSSSSNSQNASPKIVVRQQSSSFDLRQQLARLGRQLASGQDGHGGISTILIIINLLL
ILLSICCDVCRSHNYTVHQSEPVSQDMRLLRPKLDSDVVEKVAIWHKHAAPPSIVEGIAISS
RPQSTMAHHPDDRDRDRDPSEEQHGVDERMVLERVTRDCVQRCIVEEDLFLDEFGIQCEKADNGEK
CYKTRCTKGCAQWYRALKELESCQEACLSLQFYPYDMPCIGACEMAQRDYWHLQRLAISHLVERTQ
PQLERAPRADGQSTPLTIRWAMHFPEHYLASRPFNIQYQFVDHGHGEELDELEDQDASGETGSSAW
FNLADYDCDEYYMCEILEALIPYTQYRFRFELPFGENRDEVLYSPATPAYQTPPEGAPISAPVIEH
LMGLDDSHLAVHWHPRFTNGPIEGYRLRLSSSEGNATSEQLVPAGRGSYIFSQQLAGTNYTLALS
MINKQGEQVAKGFVQTHSARNEKPAKDLTESVLLVGRRAVMWQSLPAGENSIMIQSQEELADIA
WSKREQQLWLLNVHGLRSLKFESGMVSPAQLKLDLGNISSGRVWPRRLSFDWLHHRLYFAMES
PERNQSSFQIISTDLLGESAKVGSFSDLPVEQLEVDALNGWIFWRNEESLWRQDLHGRMIHRLLR
IRQPGWFLVQPQHFIIHLMLPQEGKFLIISYDGGFKHPLPLPPPSNGAGNGPASSHWQSFALLGRS
LLLPPDSGQLILVEQQGQAASPSASWPLKNLPDCWAVILLVPESQPLTSAGGKPHSLKALLGAQAAK
ISWKEPERNPYQSADAARSWSYELEVLVDASQSAFISIRNIRGPIFGLQRLQPDNLYQLRVRAINVD
GEPGEWTEPLAARTWPLGPHRLRWASRQGSVIHTNELGEGLEVQEQLERLPGPMTMVNESVGYVY
TGDGLLHCINLVHSQWGCPISEPLQHVGSVTYDWRGGRVYWDLARNCVVRMDPWSGSRLELPPVE
ANFLALDPRQGHLYYATSSQLSRHGSTPDEAVTYRVRNGLEGSIAFVLDTQQDQLFWLVKGSAL
RLYRAPLTAGGDSLQMIQIKGVFQAVPDSLQLLRPLGALLWLEERSGRRARLVRLAALPDMELPT
PDQASPASALQLLDPQLPPRDEGVIPTMVLPSVRLDDGHWDFFHVRWPSTSGGNHSVSYRLLL
EFGQLQTLDLSTPFARLTQLPQAQLQLKISITPRTAWRSGDTRVQLTTPPVAPSQRRLRVFVE
RLATALQEANVSAVLRWDAPEQGEAPMQALEYHISCWVGSLEHEELRLNQSALAEARVEHLQPDQT
YHFQVEARVAATGAAAGAASHALHVAPEVQAVPRVLYANAEIFIGELDLDRNRRLVHTASPVEHL
VGIERGEQLLWVNEHVELLTHVPGSAPAKLARMRAEVLALAVDWIQRIVYWAELDATAPQAAIYR
LDLCNFEGKILQGERVWSTPRGRLLKDLVALPQAQSLIWLEYEQGSPRNGSLRGRNLDGSELEWA
TVQPLIRLHAGSLEPGSETLNLVDNQKLCVYDVARQLCTASALRAQLNLLGEDSIAQQLAQDSGY
LYAVKNWSIRAYGRRRQLEYTVELEPEEVRLLQAHNYQAYPPKNCLLLPSSGGSLKATDCEEQR
CLLNLPMITASEDCPLIPGVRYQLNLTARGPGSEHHDHVEPLGQWLLGAGESLNLTDLLPFTR
YRVSGILSSFYQKKLALPTLVLAPLELLTASATPSPRNFSVRVLSPRELEVSWLPPEQLRSESY
YTLHWQQELDGENVDQRREWEAHERRELETAGTHRLTGKPGSGYSLWVQAHAATPKSNSSERLHVR
SFAELPELQLELGPYLSLWTAGTPDPLGSLQLECRSSAEQLRRNVAGNHTKMVVEPLQPRTRYQ
CRLLLGYAATPGAPLYHGTAEVYETLGDAPSQPGKPLEHIAEEVFRVTWAARGNGAPIALYNLE
ALQARSDIRRRRRRRRRSSGSLQLPWAEPEVVVEDQWDFCNTTELSICVKSLSHSSRLLFRVR
ARSLEHGWPYSEESERVAEPFVSPEKRGSLVLAIIAPAAIVSSCVLALVLRKVQKRRRAKLL
QQSRPSIWSNLSTLQTQQQLMAVRNRAFSTTLDADIALLPQINWSQLKLLRFLGSGAFGEVYEGQ
LKTEDSEEPQRVAIKSLRKGASEFAELLQEAQLMSNFKHENIVRLVVICFDTESISLIMEHMEAGD
LLSYLRAARATSTQEPQPTAGLSSELLAMCIDVANGCSYLEDMHFVHRDLACRNCLVTSTGSTD
RRRTVKIGDFGLARDIYKSDYYRKEGEGLLPVRWMSPELVLDGLFTTQSDVWAFGLCWEILT LGQ
QPYAARNFVLAHVKEGGRLQPPMCTEKLYSLLLLCWRTDPWERPSFRRCYNTLHAISTDLRRT
QMASATADTVVSSSRPEFKVRFDGQPLEEHREHNERPEDENLTLEVPKDKQLYANEGVSRL'

```

Abbildung 26. Beispielabfrage auf Grundlage von 7LES\_DROME-Daten

Der HMMER-Wrapper setzt die Ergebnisse der Abfrage in die in Tabelle 69 gezeigte relationale Tabelle um.

Tabelle 69. Die HMMER-Ergebnisse werden in eine relationale Tabelle umgesetzt

Model	ModelScore	DomainNumber	DomainScore
pkinase	+3.04100000000000E+002	1	+3.04100000000000E+002
fn3	+1.76300000000000E+002	1	+4.90000000000000E+001
fn3	+1.76300000000000E+002	2	+1.36000000000000E+001
fn3	+1.76300000000000E+002	3	+1.62000000000000E+001
fn3	+1.76300000000000E+002	4	+6.35000000000000E+001
fn3	+1.76300000000000E+002	5	+1.46000000000000E+001
fn3	+1.76300000000000E+002	6	+1.94000000000000E+001
rrm	-4.45000000000000E+001	1	-4.45000000000000E+001

Die Daten weisen nun ein relationales Format auf und können mit Daten von anderen Datenquellen verknüpft werden.

**Zugehörige Tasks:**

- „Hinzufügen von HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 290

---

## Hinzufügen von HMMER zu einem Server mit zusammengeschlossenen Datenbanken

### Hinzufügen von HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf HMMER-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken auf HMMER-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

**Voraussetzungen:**

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.

**Vorgehensweise:**

Gehen Sie wie folgt vor, um HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Stellen Sie sicher, dass die korrekte Version der ausführbaren HMMER-Programmdateien installiert ist.
2. Konfigurieren Sie den HMMER-Dämon.
3. Starten Sie den HMMER-Dämon.
4. Registrieren Sie den Wrapper.
5. Registrieren Sie die Serverdefinitionen.
6. Registrieren Sie den Kurznamen.

**Zugehörige Konzepte:**

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Was ist HMMER?“ auf Seite 287

**Zugehörige Tasks:**

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Überprüfen der Version der ausführbaren HMMER-Programmdatei“ auf Seite 291
- „Konfigurieren des HMMER-Dämons“ auf Seite 291

- „Registrieren des HMMER-Wrappers“ auf Seite 299
- „Registrieren der Serverdefinition für eine HMMER-Datenquelle“ auf Seite 300
- „Registrieren von Kurznamen für HMMER-Datenquellen“ auf Seite 302
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43

#### Zugehörige Referenzen:

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „HMMER-Datenquelle - vollständiges Beispiel“ auf Seite 308

## Überprüfen der Version der ausführbaren HMMER-Programmdatei

Die Prüfung, ob die erforderliche Version der ausführbaren HMMER-Programmdatei installiert ist, gehört zu der umfangreicheren Task des Hinzufügens von HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Auf dem Server, auf dem das HMMER-Anwendungsprogramm installiert ist, muss eine unterstützte Version der ausführbaren Dateien 'hmpfam' und 'hmmsearch' installiert sein. Das HMMER-Anwendungsprogramm kann entweder auf dem Server mit zusammengeschlossenen Datenbanken oder auf einem separaten HMMER-Server installiert werden.

#### Vorgehensweise:

Gehen Sie wie folgt vor, um den Versionsstand der ausführbaren Datei zu prüfen:

1. Geben Sie einen Befehl aus, der die Versionsnummer zurückgibt.
  - Dieser Befehl lautet für das Programm 'hmpfam':  
`hmpfam -h`
  - Dieser Befehl lautet für das Programm 'hmmsearch':  
`hmmsearch -h`
2. Prüfen Sie in der Ausgabedatei die Version der ausführbaren Dateien. Sie müssen über HMMER Version 2.2g (oder höher) verfügen.
3. Steht Ihnen die korrekte Version nicht zur Verfügung, laden Sie die Dateien unter folgender Adresse herunter: <http://hmmmer.wustl.edu/>.

Die nächste Task in dieser Tasksequenz ist das Konfigurieren des HMMER-Dämons.

#### Zugehörige Tasks:

- „Konfigurieren des HMMER-Dämons“ auf Seite 291

## Konfigurieren des HMMER-Dämons

Das Konfigurieren des HMMER-Dämons ist Teil der umfangreicheren Task des Hinzufügens von HMMER zu einem Server mit zusammengeschlossenen Datenbanken.



Der HMMER-Wrapper erfordert einen HMMER-Dämon. Dieser HMMER-Dämon muss auf einem Server ausgeführt werden, auf den Sie von Ihrem System zusammengeschlossener DB2 UDB-Datenbanken aus über TCP/IP zugreifen können. Dabei kann dies derselbe Server sein, der als Server mit zusammengeschlossenen Datenbanken betrieben wird, oder es kann ein separater HMMER-Server sein.

Der Dämon wird getrennt vom Wrapper und DB2 Universal Database ausgeführt. Er ist empfangsbereit für HMMER-Jobanforderungen vom Wrapper.

#### **Voraussetzungen:**

Der HMMER-Dämon muss über Folgendes verfügen:

- Ausführungszugriff auf die ausführbaren Dateien `hmpfam` und `hmmsearch`, sodass HMMER-Suchen ausgeführt werden können.
- Schreibzugriff auf ein Verzeichnis, in das temporäre Dateien geschrieben werden können.
- Lesezugriff auf mindestens eine Profildatenbank, für die HMMER-Suchen ausgeführt werden können.

#### **Einschränkungen:**

Der HMMER-Dämon wird unter Umständen nicht korrekt ausgeführt, wenn der Pfad der ausführbaren Datei oder der Datenbank Leerzeichen enthält. Sie sollten beispielsweise die ausführbare HMMER-Datei auf Windows-Servern nicht unter `C:\Program Files` installieren.

#### **Vorgehensweise:**

Gehen Sie wie folgt vor, um den HMMER-Dämon zu konfigurieren:

1. Stellen Sie sicher, dass die ausführbaren Dateien des HMMER-Dämons sich auf dem korrekten Server befinden. Während der Installation von DB2 Information Integrator werden die ausführbaren Dämondateien in einem Verzeichnis auf dem Server mit zusammengeschlossenen Datenbanken installiert:

##### **Unter UNIX:**

Die ausführbaren Dämondateien werden im Verzeichnis `$DB2PATH/bin` installiert.

##### **Unter Windows:**

Die ausführbaren Dämondateien werden im Verzeichnis `%DB2PATH%\bin` installiert.

Wenn Sie einen separaten HMMER-Server verwenden, müssen Sie die ausführbaren Dämondateien von dem Verzeichnis auf dem Server mit zusammengeschlossenen Datenbanken in ein Verzeichnis auf dem HMMER-Server kopieren. Die ausführbaren Dämondateien können in einem beliebigen Verzeichnis auf dem HMMER-Server ausgeführt werden, wenn der Name im Verzeichnispfad keine Leerzeichen enthält.

2. Stellen Sie sicher, dass die Konfigurationsdatei sowie andere erforderliche Dateien sich auf dem Server befinden, auf dem HMMER installiert ist. Einige der erforderlichen Dateien sind zusammen mit DB2 Information Integrator in einem Verzeichnis auf dem Server mit zusammengeschlossenen Datenbanken installiert. Die anderen erforderlichen Dateien müssen Sie bereitstellen.

##### **Unter UNIX:**

Folgende Dateien müssen sich auf dem Server befinden, auf dem HMMER installiert ist:

- Die ausführbare Dämondatei: %DB2PATH%/bin/db2hmmmer\_daemon
  - Die Konfigurationsdatei des HMMER-Dämons:  
%DB2PATH%/samples/lifesci/HMMER\_DAEMON.config
  - Das Konvertierungsdienstprogramm: %DB2PATH%/bin/db2h2x
  - Die Shellprozedur: %DB2PATH%/bin/db2runpfam.ksh
  - Die ausführbaren HMMER-Dateien (nicht durch IBM bereitgestellt):  
hmpfam und hmmsearch
  - Die HMMER-Datenbankdateien (nicht durch IBM bereitgestellt)
- %DB2PATH% ist der Pfad, in dem DB2 Information Integrator installiert ist.

#### Unter Windows:

Folgende Dateien müssen sich auf dem Server befinden, auf dem HMMER installiert ist:

- Die ausführbaren Dämondateien:  
%DB2PATH%\bin\db2hmmmer\_daemon.exe und  
%DB2PATH%\bin\db2hmmmer\_daemon\_svc.exe
- Die Konfigurationsdatei des HMMER-Dämons:  
%DB2PATH%\samples\lifesci\HMMER\_DAEMON.config
- Das Konvertierungsdienstprogramm: %DB2PATH%\bin\db2h2x.exe
- Die ausführbaren HMMER-Dateien (nicht durch IBM bereitgestellt):  
hmpfam.exe und hmmsearch.exe
- Die HMMER-Datenbankdateien (nicht durch IBM bereitgestellt)

%DB2PATH% ist der Pfad, in dem DB2 Information Integrator installiert ist.

Der Dämon sucht die Konfigurationsdatei standardmäßig in dem Arbeitsverzeichnis, von dem aus der Dämon gestartet wird. Sie können die Konfigurationsdatei in eine andere Position kopieren. Wenn Sie einen HMMER-Server verwenden, müssen Sie die Dämonkonfigurationsdatei von dem Verzeichnis auf dem Server mit zusammengeschlossenen Datenbanken in ein Verzeichnis auf dem HMMER-Server kopieren. Sie können die Dämonkonfigurationsdatei in ein beliebiges Verzeichnis auf dem HMMER-Server kopieren, auf das der Dämon zugreifen kann.

3. Stellen Sie unter UNIX sicher, dass sich die ausführbare HMMER-Dämondatei, das Konvertierungsdienstprogramm und die Shellprozedur ausführen lassen. Um die Dateien ausführbar zu machen, geben Sie folgenden Befehl aus:  
chmod a+x db2hmmmer\_daemon db2h2x db2runpfam.ksh
4. Bearbeiten Sie die Dämonkonfigurationsdatei so, dass sie mit Ihrer Datenquelle ausgeführt werden kann. Sie können die Konfigurationsdatei auch umbenennen.
  - Die erste Zeile in der Konfigurationsdatei muss aus einem Gleichheitszeichen bestehen. Fehlt das Gleichheitszeichen, wird der Dämon nicht gestartet. Eine Fehlermeldung meldet dann, dass der DAEMON\_PORT nicht angegeben wurde.
  - Die letzte Zeile in der Konfigurationsdatei muss durch eine neue Zeile abgeschlossen werden. Die mit DB2 Information Integrator bereitgestellte Beispielformatkonfigurationsdatei endet mit einer neuen Zeile. Stellen Sie beim Bearbeiten der Datei sicher, dass die letzte Zeile in der Datei stets mit einer neuen Zeile abgeschlossen wird. Ist dies nicht der Fall, erhalten Sie eine Fehlermeldung bei dem Versuch, die erste HMMER-Abfrage unter Verwendung der in der letzten Zeile aufgeführten Datenquelle auszuführen.

- Geben Sie die folgenden Optionen in der Konfigurationsdatei an. Bei Optionen, die einen Pfad erfordern, können relative Pfade angegeben werden. Relative Pfade beziehen sich auf das Verzeichnis, über das der Dämonprozess gestartet wurde.

#### **DAEMON\_PORT**

Hierbei handelt es sich um den Netzport, an dem der Dämon HMMER-Jobanforderungen empfängt, die vom Wrapper übergeben werden.

#### **MAX\_PENDING\_REQUESTS**

Hierbei handelt es sich um die maximale Anzahl an HMMER-Jobanforderungen, die für den Dämon zu einem bestimmten Zeitpunkt geblockt werden können. Es geht hier nicht um die Anzahl der HMMER-Jobs, die gleichzeitig ausgeführt werden, sondern lediglich um die Anzahl der Jobanforderungen, die zu einem bestimmten Zeitpunkt geblockt werden können. Es wird empfohlen, für diese Option einen Wert von über 5 anzugeben. Für den HMMER-Dämon gibt es keine Einschränkungen hinsichtlich der Anzahl der HMMER-Jobs, die gleichzeitig ausgeführt werden können.

#### **DAEMON\_LOGFILE\_DIR**

Hierbei handelt es sich um das Verzeichnis, in dem der Dämon die ihm zugeordnete Protokolldatei erstellt. Diese Datei enthält nützliche Status- und Fehlerinformationen, die vom HMMER-Dämon generiert werden.

#### **Q\_SEQ\_DIR\_PATH**

Hierbei handelt es sich um das Verzeichnis, in dem der Dämon eine temporäre Datendatei mit der Abfragesequenz erstellt. Diese temporäre Datei wird nach der Beendigung eines HMMER-Jobs bereinigt.

#### **HMMER\_OUT\_DIR\_PATH**

Hierbei handelt es sich um das Verzeichnis, in dem der Dämon die temporäre Datei zur Speicherung der HMMER-Ausgabedaten erstellt. Die Daten werden aus dieser Datei gelesen und über die Netzwerkverbindung zurück an den Wrapper übergeben. Nach der Übergabe der Daten an den Wrapper bereinigt der Dämon die temporäre Datei.

#### **RUNPFAM\_PATH**

Dies ist der vollständig qualifizierte Name der Shellprozedur db2runpfam.ksh, die durch DB2 Information Integrator bereitgestellt wird. Diese Option wird bei Angabe unter Windows ignoriert.

#### **HMMERPFAM\_PATH**

Hierbei handelt es sich um den vollständig qualifizierten Namen der ausführbaren HMMER-Datei auf der Maschine, die den Dämon ausführt. Unter UNIX lautet der Name der Datei hmmpfam. Unter Windows lautet der Name der Datei hmmpfam.exe.

#### **HMMSEARCH\_PATH**

Hierbei handelt es sich um den vollständig qualifizierten Namen der ausführbaren HMMER-Datei auf der Maschine, die den Dämon ausführt. Unter UNIX lautet der Name der Datei hmmsearch. Unter Windows lautet der Name der Datei hmmsearch.exe.

#### **H2X\_PATH**

Dies ist der vollständig qualifizierte Name des mit dem Dämon bereitgestellten Konvertierungsprogramms (Konvertierung von HMMER in XML). Unter UNIX lautet der Name des Programms db2h2x. Unter Windows lautet der Name des Programms db2h2x.exe.

### Datenbankspezifikationseintrag

Dieser Eintrag gibt die Speicherposition einer Profildatenbank oder Sequenzdatei an. Notieren Sie sich die Datenbank *datenquellenname*, die Sie in der Konfigurationsdatei angeben. Damit der Dämon ordnungsgemäß funktioniert, müssen Sie beim Erstellen des Kurznamens für die Datenquelle die Datenbank *datenquellenname* angeben. Der Name ist abhängig von der Groß-/Kleinschreibung. Die Datenbank *datenquellenname* wird wie folgt angegeben:

- In der Option DATASOURCE der Anweisung CREATE NICKNAME (für 'hmmpfam')
- Im Vergleichselement MODEL der Anweisung CREATE NICKNAME (für 'hmmsearch')

Die Konfigurationsdatei muss mindestens einen Datenbankspezifikationseintrag in folgendem Format enthalten:

```
datenquellenname=vollständig_qualifizierter_name_der_profildatenbank_oder_sequenzdatenbank
```

#### Unter UNIX:

Um beispielsweise die Profildatenbank MYHMMS anzugeben, fügen Sie der Konfigurationsdatei des Dämons die folgende Zeile hinzu:

```
myhmms=/home/user_ID/myhmms
```

#### Unter Windows:

Um beispielsweise die Profildatenbank MYHMMS anzugeben, fügen Sie der Konfigurationsdatei des Dämons die folgende Zeile hinzu:

```
myhmms=c:\hmmmer\tutorial\myhmms
```

Die nächste Task in dieser Tasksequenz ist das Starten des HMMER-Dämons.

#### Zugehörige Tasks:

- „Starten des HMMER-Dämons“ auf Seite 296

#### Zugehörige Referenzen:

- „Konfigurationsdatei für den HMMER-Dämon - Beispiele“ auf Seite 295

## Konfigurationsdatei für den HMMER-Dämon - Beispiele

Folgende Beispiele zeigen den Inhalt einer Beispielkonfigurationsdatei für PFAM und SEARCH.

#### Beispiel - Datei HMMER\_DAEMON.config für UNIX:

Dieses Beispiel zeigt die erforderlichen Optionen und Spezifikationen für die Profildatenbank für UNIX.

```
=
DAEMON_PORT=4098
MAX_PENDING_REQUESTS=10
DAEMON_LOGFILE_DIR=./
Q_SEQ_DIR_PATH=./
HMMER_OUT_DIR_PATH=./
RUNPFAM_PATH=./db2runpfam.ksh
HMMPFAM_PATH=/home/user_id/hmmer/bin/hmmpfam
HMMSEARCH_PATH=/home/user_id/hmmer/bin/hmmsearch
```

```
H2X_PATH=/home/user_id/sqllib/bin/db2h2x
myhmms=/home/user_id/hmmer/tutorial/myhmms
globin=/home/user_id/hmmer/tutorial/globin.hmm
pfamls=/home/user_id/hmmer/pfam/Pfam_ls
```

### Beispiel - Datei HMMER\_DAEMON.config für Windows:

Dieses Beispiel zeigt die erforderlichen Optionen und Spezifikationen für Windows.

```
=
DAEMON_PORT=4098
MAX_PENDING_REQUESTS=10
DAEMON_LOGFILE_DIR=.\
Q_SEQ_DIR_PATH=.\
HMMER_OUT_DIR_PATH=.\
HMMPFAM_PATH=c:\hmmer\bin\hmpfam.exe
HMMSEARCH_PATH=c:\hmmer\bin\hmmsearch.exe
H2X_PATH=.\db2h2x.exe
myhmms=c:\hmmer\tutorial\myhmms
globin=c:\hmmer\tutorial\globin.hmm
pfamseq=c:\hmmer\pfam\pfamseq
```

### Zugehörige Tasks:

- „Konfigurieren des HMMER-Dämons“ auf Seite 291

## Starten des HMMER-Dämons

Das Starten des HMMER-Dämons ist ein Teil der umfangreicheren Task des Hinzufügens von HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken. Bevor Sie auf HMMER-Datenquellen zugreifen können, müssen Sie den HMMER-Dämon starten.

### Voraussetzungen:

Bevor Sie den HMMER-Dämon starten, müssen Sie über Schreibzugriff auf alle Pfade verfügen, die unter den Einträgen DAEMON\_LOGFILE\_DIR, HMMER\_OUT\_DIR\_PATH und Q\_SEQ\_DIR\_PATH in der Konfigurationsdatei aufgeführt sind.

### Vorgehensweise:

Gehen Sie wie folgt vor, um den HMMER-Dämon auf einem UNIX-Server zu starten:

1. Öffnen Sie das Verzeichnis, in dem die ausführbare Dämondatei sich befindet.
2. Geben Sie den Befehl `db2hmmer_daemon` aus.
  - Wenn Sie den Namen der Dämonkonfigurationsdatei nicht geändert haben und sich die Konfigurationsdatei in demselben Verzeichnis befindet wie die ausführbare Dämondatei, geben Sie den folgenden Befehl über die Befehlszeile ein:  
`db2hmmer_daemon`
  - Falls Sie den Namen der Dämonkonfigurationsdatei geändert haben oder sich die Dämonkonfigurationsdatei nicht in demselben Verzeichnis befindet wie die ausführbare Dämondatei, verwenden Sie im Dämonbefehl des Wrappers die Option `'-c'`, um die ausführbare Dämondatei auf den neuen Namen bzw. die neue Position zu verweisen.

Wird beispielsweise der folgende Befehl ausgegeben, sucht der Wrapperdämon die Konfigurationsdaten in einer Datei namens HMMER\_D.config im Unterverzeichnis cfg.

```
db2hmmmer_daemon -c cfg/HMMER_D.config
```

Die ausführbare Datei startet einen neuen Prozess, in dem der HMMER-Dämon ausgeführt wird.

Gehen Sie wie folgt vor, um den Dämon auf einem UNIX-Server zu stoppen:

1. Listen Sie die Prozess-ID des Dämons `db2hmmmer_daemon` mit Hilfe des folgenden UNIX-Befehls auf:

```
ps -ef | grep db2hmmmer
```

2. Stoppen Sie den Dämon mit Hilfe der Prozess-ID. Verwenden Sie dazu diesen Befehl:

```
kill nnnn
```

Dabei ist `nnnn` die Prozess-ID des Dämons `db2hmmmer_daemon`.

Gehen Sie wie folgt vor, um den HMMER-Dämon auf einem Windows-Server zu starten:

1. Öffnen Sie das Verzeichnis, in dem die ausführbare Dämondatei sich befindet.
2. Geben Sie den Befehl `db2hmmmer_daemon` mit den erforderlichen Parametern aus. Um beispielsweise den Dämonservice mit aktiviertem Debug zu installieren und den Dämon zu starten, geben Sie folgende Befehle aus:

```
db2hmmmer_daemon -a install -d 2
```

```
db2hmmmer_daemon -a start
```

Um den Dämon zu stoppen, verwenden Sie den folgenden Windows-Befehl:

```
db2hmmmer_daemon -a stop
```

Die nächste Task in dieser Tasksequenz ist das Registrieren des HMMER-Wrappers.

#### Zugehörige Tasks:

- „Konfigurieren des HMMER-Dämons“ auf Seite 291
- „Registrieren des HMMER-Wrappers“ auf Seite 299
- „Hinzufügen von HMMER-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 290

#### Zugehörige Referenzen:

- „Befehl `db2hmmmer_daemon` - Syntax und Beispiele“ auf Seite 297

## Befehl `db2hmmmer_daemon` - Syntax und Beispiele

Der Befehl `db2hmmmer_daemon` kann auf UNIX- und Windows-Servern verwendet werden. Einige der in der Syntax aufgelisteten Argumente können nur auf Windows-Servern verwendet werden.

Die Syntax für den Befehl `db2hmmmer_daemon` lautet:

```
db2hmmmer_daemon -a aktion -c konfigurationsdatei -d debugstufe  
-u benutzer-id -p kennwort
```

**-a *aktion***

Führt die angegebene Aktion aus. Gültige Aktionen sind *status*, *install*, *start*, *stop* und *remove*.

Sie können dieses Argument nur auf Windows-Servern angeben.

**-c** *konfigurationsdatei*

Weist den Dämonservice zur Verwendung der angegebenen Konfigurationsdatei anstelle der Standardkonfigurationsdatei an. Geben Sie die Konfigurationsdatei nicht an, sucht der Dämon die Datei `HMMER_DAEMON.config` in dem Verzeichnis, in dem die ausführbaren Dämondateien installiert sind. Sie können diese Option mit den Aktionen *install* und *start* verwenden.

Dieses Argument kann für UNIX- und Windows-Server angegeben werden.

**-d** *debugstufe*

Setzt die Debugstufe des Dämonservice auf den angegebenen Wert. Gültige Werte sind 1, 2 oder 3. Sie können diese Option mit den Aktionen *install* und *start* verwenden.

Dieses Argument kann für UNIX- und Windows-Server angegeben werden.

**-u** *benutzer-id*

Legt für den Service die Ausführung unter der angegebenen Benutzer-ID fest. Sie können diese Option mit der Aktion *install* verwenden.

Sie können dieses Argument nur auf Windows-Servern angeben.

**-p** *kennwort*

Gibt das Kennwort für die angegebene Benutzer-ID an. Das Kennwort ist nur dann gültig und erforderlich, wenn Sie die Option '-u' angeben. Ist beim Festlegen der Option '-u' die Option '-p' nicht angegeben, fordert Sie das Programm zur Eingabe des Kennworts auf. Sie können diese Option mit der Aktion *install* verwenden.

Sie können dieses Argument nur auf Windows-Servern angeben.

Die mit der Aktion *start* angegebenen Optionen wirken sich nur auf die aktuelle Ausführung des Dämons aus und überschreiben die Werte, die mit der Aktion *install* angegeben werden.

**Beispiele:**

Die folgenden Beispiele zeigen Dämonaktionen unter Windows. In diesen Beispielen wird davon ausgegangen, dass die Datei `HMMER_DAEMON.config` sich in demselben Verzeichnis befindet wie die `db2hmmmer_daemon.exe`.

- Geben Sie Folgendes ein, um den Status des Dämons zu prüfen:  
`db2hmmmer_daemon -a status`
- Geben Sie Folgendes ein, um den Dämonservice mit aktiviertem Debug zu installieren:  
`db2hmmmer_daemon -a install -d 2`
- Geben Sie Folgendes ein, um den Dämon zu starten:  
`db2hmmmer_daemon -a start`
- Geben Sie Folgendes ein, um den Dämon zu stoppen:  
`db2hmmmer_daemon -a stop`
- Geben Sie Folgendes ein, um den Dämonservice zu entfernen oder zu deinstallieren:  
`db2hmmmer_daemon -a remove`

**Zugehörige Tasks:**

- „Starten des HMMER-Dämons“ auf Seite 296



## Registrieren des HMMER-Wrappers

Die Registrierung des HMMER-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf XML-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Bibliotheksdatei des Wrappers aus.

Um beispielsweise einen Wrapper mit dem Namen `hmmwr_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER hmmwr_wrapper LIBRARY 'libdb2lshmmwr.a';
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung CREATE WRAPPER finden Sie in der Liste der Bibliotheksdateien des HMMER-Wrappers.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinition für den HMMER-Wrapper.

### Zugehörige Tasks:

- „Registrieren der Serverdefinition für eine HMMER-Datenquelle“ auf Seite 300

### Zugehörige Referenzen:

- „Bibliotheksdateien für den HMMER-Wrapper“ auf Seite 299
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien für den HMMER-Wrapper

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den HMMER-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, werden 3 Bibliotheksdateien zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden folgende Wrapperbibliotheksdateien zum Verzeichnispfad hinzugefügt: `libdb2lshmmwr.a`, `libdb2lshmmwrF.a` und `libdb2lshmmwrU.a`.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.



Tabelle 70. Positionen und Dateinamen von HMMER-Wrapperbibliotheken

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2lshmmmer.a
Linux	/opt/IBM/db2/V8.1/lib	libdb2lshmmmer.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lshmmmer.so
Windows	%DB2PATH%\bin	db2lshmmmer.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist C:\Program Files\IBM\SQLLIB.

#### Zugehörige Tasks:

- „Registrieren des HMMER-Wrappers“ auf Seite 299

## Registrieren der Serverdefinition für eine HMMER-Datenquelle

Die Registrierung der Serverdefinitionen für HMMER-Datenquellen ist ein Teil der umfangreicheren Task des Hinzufügens von HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Nach der Registrierung des Wrappers muss eine entsprechende Serverdefinition registriert werden.

#### Vorgehensweise:

Um die HMMER-Serverdefinition zu registrieren, geben Sie die Anweisung CREATE SERVER aus.

Beispiel:

```
CREATE SERVER server_name TYPE HMMER_search_type
  VERSION version WRAPPER wrapper_name
  OPTIONS (NODE 'node_name', DAEMON_PORT 'port_number')
```

Für jeden Server, für den eine HMMER-Suche ausgeführt werden soll, muss eine Definition registriert werden.

Die nächste Task in dieser Tasksequenz ist das Registrieren von Kurznamen für HMMER-Datenquellen.

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für HMMER-Datenquellen“ auf Seite 302

#### Zugehörige Referenzen:

- Anhang D, „Serveroptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 641
- „Anweisung CREATE SERVER - Beispiele für HMMER-Wrapper“ auf Seite 300

## Anweisung CREATE SERVER - Beispiele für HMMER-Wrapper

Dieser Abschnitt enthält ein Beispiel, in dem die Verwendung der Anweisung CREATE SERVER zur Registrierung von Serverdefinitionen für den HMMER-Wrapper gezeigt wird.

Um die Serverdefinition `hmmpfam_server` für Suchen mit dem Programm `hmmpfam` zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE SERVER hmmpfam_server
  TYPE pfam
  VERSION 2.2
  WRAPPER hmmer_wrapper
  OPTIONS (NODE 'someserver.someschool.edu', DAEMON_PORT '4098')
```

#### *hmmpfam\_server*

Ein Name, der der HMMER-Serverdefinition zugeordnet wird. Dieser Name muss eindeutig sein.

#### **TYPE** *pfam*

Der Typ der von der Serverdefinition unterstützten Suche. Als Typen können PFAM (für `hmmpfam`) oder SEARCH (für `hmmsearch`) angegeben werden.

#### **VERSION** 2.2

Die Version der ausführbaren Datei `hmmpfam` oder `hmmsearch`, die Sie verwenden. Unterstützte Versionen sind HMMER 2.2g (oder höher).

#### **WRAPPER** *hmmer\_wrapper*

Der Name, den Sie in der Anweisung `CREATE WRAPPER` angegeben haben.

#### **NODE** *'someserver.someschool.edu'*

Der Hostname oder die IP-Adresse des Servers, auf dem der HMMER-Dämonprozess ausgeführt wird.

Obwohl der Knotenname als Option in der Anweisung `CREATE SERVER` angegeben wird, ist er für HMMER-Datenquellen erforderlich.

#### **DAEMON\_PORT** *'4098'*

Die Nummer des Ports, an dem der Dämon HMMER-Jobanforderungen empfängt. Die Anschlussnummer muss der Nummer entsprechen, die in der Option `'DAEMON_PORT'` der Dämonkonfigurationsdatei angegeben wurde. Der Standardwert lautet 4098.

#### **Zusätzliche Serveroptionen:**

Beim Erstellen der Serverdefinition können Sie in der Anweisung `CREATE SERVER` zusätzliche Serveroptionen angeben.

#### **PROCESSORS**

Die Anzahl der Prozessoren, die das Programm HMMER verwendet. Diese Option entspricht der Option `--cpu` der Befehle `hmmpfam` und `hmmsearch`. Beispiel: `PROCESSORS '2'`.

#### **HMMPFAM\_OPTIONS**

Mit Hilfe dieser Serveroption können Sie Optionen an den Befehl `hmmpfam` übergeben, die nicht in einem Vergleichselement angegeben werden können. Beispiel: `HMMPFAM_OPTIONS '--null12 --pvm'`. In diesem Beispiel wird `hmmpfam` bei jeder Abfrage für diesen Server mit den beiden zusätzlichen Optionen `--null12` und `--pvm` ausgeführt. Die Option `HMMPFAM_OPTIONS` ist nur für Server gültig, die als Typ PFAM angegeben sind.

#### **HMMSEARCH\_OPTIONS**

Mit Hilfe dieser Serveroption können Sie Optionen an den Befehl `hmmsearch` übergeben, die nicht in einem Vergleichselement angegeben werden können. Beispiel: `HMMSEARCH_OPTIONS '--null12 --pvm'`. In diesem Beispiel wird `hmmsearch` bei jeder Abfrage für diesen Server mit den beiden zusätz-

lichen Optionen `--null2` und `--pvm` ausgeführt. Die Option `HMMSEARCH_OPTIONS` ist nur für Server gültig, die als Typ `SEARCH` angegeben sind.

**Zugehörige Tasks:**

- „Registrieren der Serverdefinition für eine HMMER-Datenquelle“ auf Seite 300

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641

## Registrieren von Kurznamen für HMMER-Datenquellen

Die Registrierung von Kurznamen für HMMER-Datenquellen ist ein Teil der umfangreicheren Task des Hinzufügens von HMMER-Datenquellen zu einem Server mit zusammenschlossenen Datenbanken.

Nach der Registrierung einer Serverdefinition muss ein entsprechender Kurzname registriert werden. Kurznamen werden in Abfragen zum Verweisen auf HMMER-Datenquellen verwendet.

**Vorgehensweise:**

Um einen HMMER-Kurznamen zu registrieren, geben Sie die Anweisung `CREATE NICKNAME` aus. Beispiel:

```
CREATE NICKNAME nickname
  FOR SERVER server_name
  OPTIONS(DATASOURCE 'data_source')
```

Sie müssen für jede Profildatenbank, die Sie abfragen wollen, einen eigenen Kurznamen definieren. Der Name von `data_source` muss mit einem vorhandenen `data_source_name` in der Datei `HMMER_DAEMON.config` auf dem HMMER-Server übereinstimmen.

Wenn Sie einen Kurznamen für eine HMMER-Datenbank erstellen, wird ein Satz von fixierten Spalten (Eingabe- und Ausgabespalten) für die Profildatenbank im Katalog des Systems zusammenschlossener Datenbanken registriert.

Dies war die letzte Task in dieser Tasksequenz.

**Zugehörige Tasks:**

- „Angeben von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75
- „Konfigurieren des HMMER-Dämons“ auf Seite 291
- „Erstellen neuer HMMER-Abfragen mit Hilfe von Beispielen“ auf Seite 309
- „Hinzufügen von HMMER-Datenquellen zu einem Server mit zusammenschlossenen Datenbanken“ auf Seite 290

**Zugehörige Referenzen:**

- „Fixierte Spalten für HMMER-Kurznamen“ auf Seite 303
- „HMMER-Datenquelle - vollständiges Beispiel“ auf Seite 308

- „Anweisung CREATE NICKNAME - Beispiel für den HMMER-Wrapper“ auf Seite 307
- „Fixierte Spalten für HMMER-Kurznamen“ auf Seite 303

## Fixierte Spalten für HMMER-Kurznamen

Wenn Sie die Anweisung CREATE NICKNAME für eine HMMER-Datenquelle ausgeben, wird automatisch ein Satz fixierter Eingabe- und Ausgabespalten mit dem Kurznamen erstellt. Wenn Sie den standardmäßigen, einer fixierten Spalte zugeordneten Datentyp ändern möchten, geben Sie den Spaltennamen und Datentyp in der Anweisung CREATE NICKNAME an. Um beispielsweise die Ausgabe der Spalte AlignmentConsensus auf die ersten 100 Zeichen zu beschränken, geben Sie folgende Anweisung aus:

```
CREATE NICKNAME nucleol (AlignmentConsensus VARCHAR(100))
FOR SERVER searchtest
OPTIONS(DATASOURCE 'nucleol', TIMEOUT '1');
```

Sie können auf die fixierten Spalten in SQL-Abfragen als Teil der Kurznamen- definition verweisen. Es gibt zwei Typen von fixierten Spalten: Eingabespalten (input) und Ausgabespalten (output).

### Fixierte Eingabespalten für HMMER-Kurznamen

Fixierte Eingabespalten werden in der Klausel WHERE angegeben. Eingabespalten werden in SQL-Abfragen als Vergleichselemente verwendet, die Parameter übergeben. Sie übergeben HMMER-Standardswitches entweder an hmmpfam oder hmmsearch. Anschließend wird HMMER für die angegebene Datenquelle ausgeführt, wobei diese Switches verwendet werden. Auf fixierte Eingabespalten kann auch in der Abfrageauswahlliste (SELECT-Liste) verwiesen werden. Diese Spalten werden als Teil der Ergebnistabelle zurückgegeben.

#### Fixierte Eingabespalten für Server des Typs PFAM:

Die folgende Tabelle listet die fixierten Spalten auf, die in der Klausel WHERE verwendet werden können.

Tabelle 71. Fixierte Eingabespalten für Server des Typs PFAM

Name	Datentyp	Beschreibung	Operator	Switches	Rückgabewert
HmmQSeq	varchar(32000)	Eingabesequenz für die Suche	=		Derselbe wie der Eingabewert, den Sie angeben. Diese Spalte ist erforderlich.
ModelEValue	double	Geschätzter E-Wert	<	-E <i>n</i>	Siehe Ausgabe.
ModelScore	double	Unformatierter Score	>	-T <i>n</i>	Siehe Ausgabe.

Tabelle 71. Fixierte Eingabespalten für Server des Typs PFAM (Forts.)

Name	Datentyp	Beschreibung	Operator	Switches	Rückgabewert
DBSize	integer	Berechnung von E-Werten, als enthielte die Datenbank 'n' Gensequenzen .	=	-Z n	Derselbe wie der Eingabewert, den Sie angeben. Bei keiner Angabe wird der Standardwert von hmmpfam verwendet.
CutMode	char(2)	Begrenzungsmodus (Cutoff-Modus); mögliche Werte sind ga, tc oder nc (Groß-/Kleinschreibung beachten).	=	--cut_ga --cut_tc --cut_nc	Derselbe wie der Eingabewert, den Sie angeben. Bei keiner Angabe wird NULL verwendet.
DomainScore	double	Domänen-Score	>	--domT n	Siehe Ausgabe.
DomainEValue	double	E-Wert für Domäne	<	--domE n	Siehe Ausgabe.
ForwardAlgorithm	char	Verwendet den Forward-Algorithmus anstelle von Viterbi; mögliche Werte sind 'Y' oder 'N'.	=	--forward	Derselbe wie der Eingabewert, den Sie angeben. 'N' ist der Standardwert.

### Fixierte Eingabespalten für Server des Typs SEARCH:

Die folgende Tabelle listet die fixierten Spalten auf, die in der Klausel WHERE verwendet werden können.

Tabelle 72. Fixierte Eingabespalten für Server des Typs SEARCH

Name	Datentyp	Beschreibung	Operator	Optionen	Rückgabewert
Model	varchar (32000)	Name des für die Suche verwendeten HMM-Profiles. Der Name muss einer der Datenquellennamen sein, die im Datenbank-spezifikationseintrag in der Datei HMMER_DAEMON.config aufgelistet sind.	=		Derselbe wie der Eingabewert, den Sie angeben. Diese Spalte ist erforderlich.
SequenceEValue	double	Geschätzter E-Wert	<	-E n	Siehe Ausgabe.

Tabelle 72. Fixierte Eingabespalten für Server des Typs SEARCH (Forts.)

Name	Datentyp	Beschreibung	Operator	Optionen	Rückgabewert
SequenceScore	double	Unformatierter Score	>	-T <i>n</i>	Siehe Ausgabe.
DBSize	integer	Berechnung von E-Werten, als enthielte die Datenbank 'n' Gensequenzen.	=	-Z <i>n</i>	Derselbe wie der Eingabewert, den Sie angeben. Bei keiner Angabe wird der Standardwert von hmmpfam verwendet.
CutMode	char(2)	Begrenzungsmodus (Cutoff-Modus); mögliche Werte sind ga, tc oder nc (Groß-/Kleinschreibung beachten).	=	--cut_ga --cut_tc --cut_nc	Derselbe wie der Eingabewert, den Sie angeben. Bei keiner Angabe wird NULL verwendet.
DomainScore	double	Domänen-Score	>	--domT <i>n</i>	Siehe Ausgabe.
DomainEValue	double	E-Wert für Domäne	<	--domE <i>n</i>	Siehe Ausgabe.
ForwardAlgorithm	char	Verwendet den Forward-Algorithmus anstelle von Viterbi; mögliche Werte sind 'Y' oder 'N'.	=	--forward	Derselbe wie der Eingabewert, den Sie angeben. 'N' ist der Standardwert.

### Fixierte Ausgabespalten für HMMER-Kurznamen

Sie können eine beliebige fixierte Ausgabespalte aus der SELECT-Liste angeben. Fixierte Ausgabespalten können ebenfalls in der WHERE-Klausel angegeben werden (als Vergleichselemente).

### Fixierte Ausgabespalten für PFAM:

Die folgende Tabelle listet die fixierten Spalten auf, die als Ausgabe für PFAM zurückgegeben werden.

Tabelle 73. Fixierte Ausgabespalten für PFAM

Name	Datentyp	Beschreibung
Model	varchar(32)	Name des Modells.

Tabelle 73. Fixierte Ausgabespalten für PFAM (Forts.)

Name	Datentyp	Beschreibung
ModelDescription	varchar(64)	Textbeschreibung des Modells.
ModelScore	double	Unformatierter Score ("Bit-Score").
ModelEValue	double	Geschätzter E-Wert.
ModelHits	integer	Anzahl der Domänen-Hits (Übereinstimmungen) in dem Modell.
DomainNumber	integer	Spezifische Domäne (innerhalb eines Modells).
SequenceFrom	integer	Anfangspunkt der Gensequenz.
SequenceFromGlobal	char	'Y', wenn das Alignment am Anfang der Gensequenz beginnt.
HmmFrom	integer	Anfangspunkt des Konsensmodells.
HmmFromGlobal	char	'Y', wenn das Alignment am Anfang des Konsensmodells beginnt.
HmmTo	integer	Endpunkt des Konsensmodells.
HmmToGlobal	char	'Y', wenn das Alignment am Ende des Konsensmodells aufhört.
DomainScore	double	Unformatierter Score ("Bit-Score") für die isolierte Domäne.
DomainEValue	double	Erwartungswert (E-Wert) für die isolierte Domäne.
AlignmentConsensus	varchar(32000)	Der HMM-Konsens. Die für den Konsens gezeigte Aminosäure ist die Aminosäure, die sich mit der größten Wahrscheinlichkeit an der Position gemäß dem HMM befindet, nicht unbedingt die Aminosäure mit dem höchsten Score.
AlignmentExactMatch	varchar(32000)	Entspricht dem Rest mit der größten Wahrscheinlichkeit im HMM.
AlignmentSubSequence	varchar(32000)	Zeigt die Gensequenz selbst.

#### Fixierte Ausgabespalten für SEARCH:

Die folgende Tabelle listet die fixierten Spalten auf, die als Ausgabe für SEARCH zurückgegeben werden.

Tabelle 74. Fixierte Ausgabespalten für SEARCH

Name	Datentyp	Beschreibung
Sequence	varchar(32)	Die Sequenz-ID.
SequenceDescription	varchar(64)	Die Textbeschreibung der Sequenz.
SequenceScore	double	Unformatierter Score ("Bit-Score").
SequenceEValue	double	Geschätzter E-Wert.
SequenceHits	integer	Anzahl der Domänen-Hits (Übereinstimmungen) in der Sequenz.
DomainNumber	integer	Spezifische Domäne (innerhalb einer Sequenz).
SequenceFrom	integer	Anfangspunkt der Gensequenz.

Tabelle 74. Fixierte Ausgabespalten für SEARCH (Forts.)

Name	Datentyp	Beschreibung
SequenceFromGlobal	char	'Y', wenn das Alignment am Anfang der Gensequenz beginnt.
HmmFrom	integer	Anfangspunkt des Konsensmodells.
HmmFromGlobal	char	'Y', wenn das Alignment am Anfang des Konsensmodells beginnt.
HmmTo	integer	Endpunkt des Konsensmodells.
HmmToGlobal	char	'Y', wenn das Alignment am Ende des Konsensmodells aufhört.
DomainScore	double	Unformatierter Score ("Bit-Score") für die isolierte Domäne.
DomainEValue	double	Erwartungswert (E-Wert) für die isolierte Domäne.
AlignmentConsensus	varchar(32000)	Der HMM-Konsens. Die für den Konsens gezeigte Aminosäure ist die Aminosäure, die sich mit der größten Wahrscheinlichkeit an der Position gemäß dem HMM befindet, nicht unbedingt die Aminosäure mit dem höchsten Score.
AlignmentExactMatch	varchar(32000)	Entspricht dem Rest mit der größten Wahrscheinlichkeit im HMM.
AlignmentSubSequence	varchar(32000)	Zeigt die Gensequenz selbst.

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für HMMER-Datenquellen“ auf Seite 302

#### Zugehörige Referenzen:

- „HMMER-Datenquelle - vollständiges Beispiel“ auf Seite 308
- „Anweisung CREATE NICKNAME - Beispiel für den HMMER-Wrapper“ auf Seite 307

## Anweisung CREATE NICKNAME - Beispiel für den HMMER-Wrapper

Dieser Abschnitt enthält ein Beispiel, in dem die Verwendung der Anweisung CREATE NICKNAME zur Registrierung eines Kurznamens für HMMER-Datenquellen gezeigt wird.

Um den Kurznamen `hmmpfam_nickname` mit der Serverdefinition `hmmpfam_server` zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE NICKNAME hmmpfam_nickname
FOR SERVER hmmpfam_server
OPTIONS(DATASOURCE 'myhmms', TIMEOUT '30')
```

*hmmpfam\_nickname*

Ein Name, der dem Kurznamen zugeordnet wird. Dieser Name muss eindeutig sein.

**SERVER** *hmmpfam\_server*

Der Name der Serverdefinition, der dieser Kurzname zugeordnet werden soll.



#### **DATASOURCE 'myhmms'**

Der Name der Datenbank, für die HMMER-Suchen ausgeführt werden sollen. Diese Datenbank muss in der HMMER-Dämonkonfigurationsdatei aufgelistet sein.

Obwohl die Datenquelle als Option in der Anweisung CREATE NICKNAME angegeben wird, ist sie für HMMER-Datenquellen erforderlich.

#### **TIMEOUT '30'**

Die maximale Zeitdauer in Minuten, die der Wrapper auf Ergebnisse vom Dämon wartet. Der Standardwert ist 60 Minuten.

#### **Zugehörige Tasks:**

- „Registrieren von Kurznamen für HMMER-Datenquellen“ auf Seite 302

#### **Zugehörige Referenzen:**

- Anhang F, „Kurznamenoptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 661

## **HMMER-Datenquelle - vollständiges Beispiel**

Dieser Abschnitt bietet ein Beispiel mit allen SQL-Anweisungen, die Sie zum Hinzufügen von HMMER-Datenquellen zu einem Server mit zusammenschlossenen Datenbanken ausgeben müssen. Dieses Beispiel zeigt ebenfalls eine Abfrage, die mit Hilfe des in den SQL-Anweisungen erstellten Kurznamens ausgeführt wird.

SQL-Anweisungen für HMMER-Datenquellen müssen spezielle Eingabevergleichselemente enthalten, die verwendet werden, um HMMER-Standardoptionen an die ausführbare Programmdatei 'hmmpfam' zu übergeben. Zu ihrer Gültigkeit muss jede an den HMMER-Wrapper übergebene Abfrage mindestens das Eingabevergleichselement `HmmQSeq` (für TYPE PFAM) oder `model` (für TYPE SEARCH) enthalten. Alle anderen Vergleichselemente sind optional.

Um eine HMMER-Abfrage für einen Kurznamen zu erstellen, geben Sie Eingabespalten in der WHERE-Klausel und Ausgabespalten in der SELECT-Liste an.

#### **Beispiel für das Programm 'hmmpfam':**

In diesem Beispiel werden ein Wrapper, eine Serverdefinition und ein Kurzname für das Programm `hmmpfam` auf einem AIX-Server mit zusammenschlossenen Datenbanken erstellt. In diesem Beispiel wird auch eine Abfrage ausgeführt, die ein Zeichenfolgeliteral für die Suchsequenz verwendet.

```
CREATE WRAPPER hmmer_wrapper
  LIBRARY 'libdb2lshmmer.a';

CREATE SERVER hmmpfam_server
  TYPE pfam VERSION 2.2
  WRAPPER hmmer_wrapper
  OPTIONS(NODE 'HMMERServ.MyCompany.com');

CREATE NICKNAME hmmpfam_nickname
  FOR SERVER hmmpfam_server
  OPTIONS(DATASOURCE 'myhmms', TIMEOUT '1');

-- Run the 7LES_DROME gene sequence on the hmmpfam_nickname
SELECT Model, substr(ModelDescription,1,50) as ModelDescription,
  ModelScore, ModelEValue, ModelHits, DomainNumber,
  SequenceFrom, SequenceTo, SequenceFromGlobal, SequenceToGlobal,
  HmmFrom, HmmTo, HmmFromGlobal, HmmToGlobal,
```

```

        DomainScore, DomainEValue,
        length(HmmQSeq) as "length(HmmQSeq)",
        length(AlignmentConsensus) as "length(AConsensus)",
        length(AlignmentMatch) as "length(AMatch)",
        length(AlignmentSubSeq) as "length(ASubSeq)",
        substr(HmmQSeq,1,64) as HmmQSeq,
        substr(AlignmentConsensus,1,64) as AlignmentConsensus,
        substr(AlignmentMatch, 1,64) as AlignmentMatch,
        substr(AlignmentSubSeq, 1,64) as AlignmentSubSeq
FROM hmmpfam_nickname
WHERE HmmQSeq =
'MTMFWQQNVVDHQSDQDKQAKGAAPT KRLNISFNVKIAVNVNTKMTTTHINQAPGTSS...';

```

### Beispiel für das Programm 'hmmsearch':

In diesem Beispiel werden ein Wrapper, eine Serverdefinition und ein Kurzname für das Programm hmmsearch auf einem Windows-Server mit zusammengeschlossenen Datenbanken erstellt.

```

CREATE WRAPPER hmmer_wrapper
  LIBRARY 'db21shmmer.dll'
  OPTIONS(DB2_FENCED 'Y');

CREATE SERVER hmmsearch_serv
  TYPE search VERSION 2.2
  WRAPPER hmmer_wrapper
  OPTIONS(NODE 'localhost');

CREATE NICKNAME artemia
  FOR SERVER hmmsearch_server
  OPTIONS(DATASOURCE 'artemia', TIMEOUT '1');

SELECT Model, Sequence, substr(SequenceDescription,1,50)
  as SequenceDescription, SequenceScore, SequenceEValue,
  SequenceHits, DomainNumber, SequenceFrom,
  SequenceTo, SequenceFromGlobal, SequenceToGlobal,
  HmmFrom, HmmTo, HmmFromGlobal, HmmToGlobal, DomainScore,
  DomainEValue,
  length(AlignmentConsensus) as "length(AConsensus)",
  length(AlignmentMatch) as "length(AMatch)",
  length(AlignmentSubSeq) as "length(ASubSeq)",
  substr(AlignmentConsensus,1,200) as AlignmentConsensus,
  substr(AlignmentMatch, 1,200) as AlignmentMatch,
  substr(AlignmentSubSeq, 1,200) as AlignmentSubSeq
FROM artemia
WHERE Model = 'globin' and DomainScore > 50;

```

### Zugehörige Tasks:

- „Registrieren von Kurznamen für HMMER-Datenquellen“ auf Seite 302
- „Erstellen neuer HMMER-Abfragen mit Hilfe von Beispielen“ auf Seite 309
- „Hinzufügen von HMMER-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 290

---

## Erstellen neuer HMMER-Abfragen mit Hilfe von Beispielen

Die folgenden HMMER-Beispielabfragen zeigen, wie Abfragen für HMMER-Datenquellen aufgebaut werden.

### Vorgehensweise:

Verwenden Sie die nachstehenden Beispiele als Anleitung für die Ausführung von Abfragen.

In diesen Abfragen beschreibt der Kurzname den Typ der HMMER-Suche und die Datenquelle. Einige Beispiele zeigen auch die Verwendung des HMMER-Wrappers mit anderen Datenquellen.

#### Abfrage 1.

```
SELECT Model, ModelScore, ModelEValue, DomainNumber, DomainScore, DomainEValue
FROM hmmpfam_nickname
WHERE HmmQSeq = 'MTMFWQQNVDHQSDEQDKQAKGAAPT KRLNISFNVKI AVNVNTKMTTTHINQ...'
```

Wird diese SQL-Abfrage ausgeführt, verwendet der Wrapper die angegebene Sequenz sowie die durch den Kurznamen definierte HMM-Datenbank, um das Programm 'hmmpfam' auszuführen. Der Wrapper gibt die in der Anweisung SELECT aufgelisteten Spalten zurück.

#### Abfrage 2.

```
SELECT Model, ModelScore, ModelEValue
FROM hmmpfam_nickname
WHERE HmmQSeq = 'MTMFWQQNVDHQSDEQDKQAKGAAPT KRLNISFNVKI AVNVNTKMTTTHINQ...'
AND ModelScore > 0
```

Bei Ausführung dieser SQL-Anweisung führt der Wrapper eine Suche hmmpfam für hmmpfam\_nickname mit der angegebenen Gensequenz durch. Zusätzlich übergibt der Wrapper die Option '-T 0' an den Befehl 'hmmpfam'. Diese Option befindet sich in der Liste der fixierten Spalten für HMMER-Kurznamen. Der Wrapper gibt die drei nach SELECT aufgelisteten Spalten zurück.

#### Abfrage 3.

```
SELECT Model, DomainNumber, DomainScore, DomainEValue
FROM hmmpfam_nickname
WHERE HmmQSeq = 'MTMFWQQNVDHQSDEQDKQAKGAAPT KRLNISFNVKI AVNVNTKMTTTHINQ...'
AND ModelEValue < 1
ORDER BY DomainScore DESC
```

Bei Ausführung dieser SQL-Anweisung führt der Wrapper eine Suche hmmpfam für hmmpfam\_nickname mit der angegebenen Gensequenz durch. Zusätzlich übergibt der Wrapper die Option '-E 1' an den Befehl hmmpfam. Diese Option befindet sich in der Liste der fixierten Spalten für HMMER-Kurznamen. Der Wrapper übergibt die vier nach SELECT aufgelisteten Spalten und sortiert das Ergebnis anhand von DomainScore vom höchsten Wert zum niedrigsten Wert.

#### Abfrage 4.

```
CREATE WRAPPER hmmer_wrapper
LIBRARY 'db2lshmmer.dll';

CREATE SERVER hmsearch_server
TYPE search VERSION 2.2
WRAPPER hmmer_wrapper
OPTIONS(NODE 'HMMERServ.MyCompany.com');

CREATE NICKNAME artemia_nickname
FOR SERVER hmsearch_server
OPTIONS(DATASOURCE 'artemia', TIMEOUT '1');

SELECT Model, Sequence, substr(SequenceDescription,1,50)
as SequenceDescription, SequenceScore, SequenceEValue,
SequenceHits, DomainNumber, SequenceFrom,
SequenceTo, SequenceFromGlobal, SequenceToGlobal,
HmmFrom, HmmTo, HmmFromGlobal, HmmToGlobal, DomainScore,
DomainEValue,
length(AlignmentConsensus) as "length(AConsensus)",
```

```

|         length(AlignmentMatch)    as "length(AMatch)",
|         length(AlignmentSubSeq)   as "length(ASubSeq)",
|         substr(AlignmentConsensus,1,200) as AlignmentConsensus,
|         substr(AlignmentMatch, 1,200)   as AlignmentMatch,
|         substr(AlignmentSubSeq, 1,200)   as AlignmentSubSeq
| FROM artemia_nickname
| WHERE Model = 'globin' and DomainScore > 50;

```

Bei Ausführung dieser SQL-Abfrage führt der Wrapper eine Suche 'hmmsearch' für die Sequenzdatei artemia mit Hilfe des durch globin angegebenen HMM durch. Zeilen mit einem DomainScore größer als 50 werden zurückgegeben, da der Wrapper die Option --domT 50 an den Befehl **hmmsearch** übergibt. Der Wrapper gibt die nach SELECT angegebenen Spalten zurück. Spaltenwerte von mehr als 200 Zeichen Länge werden abgeschnitten. Nur die ersten 200 Zeichen dieser Spalten werden zurückgegeben.

#### Zugehörige Tasks:

- „Hinzufügen von HMMER-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 290

#### Zugehörige Referenzen:

- „Fixierte Spalten für HMMER-Kurznamen“ auf Seite 303
- „HMMER-Datenquelle - vollständiges Beispiel“ auf Seite 308
- „Nachrichten für den HMMER-Wrapper“ auf Seite 311

---

## Nachrichten für den HMMER-Wrapper

Damit der HMMER-Wrapper funktioniert, müssen Sie eine Abfrage angeben, die ein Vergleichselement für die Spalte 'HmmQSeq' enthält. Wenn Sie ein Fragment abfragen, in dem ein Vergleichselement für die Spalte 'HmmQSeq' fehlt, erhalten Sie eine Fehlermeldung.

In diesem Abschnitt sind die Nachrichten und ihre zugehörigen Beschreibungen aufgeführt, die bei der Verwendung des HMMER-Wrappers angezeigt werden können.

*Tabelle 75. Nachrichten für den HMMER-Wrapper*

Fehlercode	Nachricht	Erläuterung
SQL0142N	Die SQL-Anweisung wird nicht unterstützt.	Die an DB2 übergebene SQL-Abfrage konnte vom Wrapper nicht verarbeitet werden. Fügen Sie das erforderliche Vergleichselement hinzu, und übergeben Sie die SQL-Anweisung erneut. Überprüfen Sie, ob der in einem Vergleichselement verwendete Operator für die betreffende Spalte gültig ist. Siehe hierzu die fixierten Spalten für HMMER-Kurznamen.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "HMMER-Wrapper" empfangen. Zugeordneter Text und Token sind "Hostname NODE kann nicht aufgelöst werden".	Der in der Anweisung CREATE SERVER angegebene TCP/IP-Name NODE ist ungültig.

Tabelle 75. Nachrichten für den HMMER-Wrapper (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "HMMER-Wrapper" empfangen. Zugeordneter Text und Token sind "Verbindung zum Dämon kann nicht hergestellt werden".	Entweder ist das Programm 'hmmmer_daemon' derzeit auf dem Zielknoten nicht aktiv oder der im Befehl CREATE SERVER angegebene Dämonanschluss (DAEMON_PORT) entspricht nicht dem Wert für DAEMON_PORT, der in der Dämonkonfigurationsdatei HMMER_DAEMON.config angegeben wurde.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "HMMER-Wrapper" empfangen. Zugeordneter Text und Token sind "Unbekannter Fehler vom HMMER-Dämon".	Der in der Anweisung CREATE NICKNAME angegebene Name für DATASOURCE entspricht keinem der Profildatenbanknamen, die in der Dämonkonfigurationsdatei HMMER_DAEMON.config aufgeführt sind.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Unspezifizierter Fehler" von der Datenquelle "HMMER-Wrapper" empfangen. Zugeordneter Text und Token sind "SCHWER WIEGENDER FEHLER: Option "--cut_TC" nicht vorhanden".	Das Vergleichselement 'CutMode' muss in Kleinbuchstaben angegeben werden. Beispiel: WHERE CutMode = 'tc'

**Zugehörige Konzepte:**

- „Nachrichten - Einführung“ in *Fehlernachrichten Band 1*

**Zugehörige Referenzen:**

- „SQLSTATE-Nachrichten“ in *Fehlernachrichten Band 2*

---

## Kapitel 15. Konfigurieren des Zugriffs auf Informix-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Informix-Datenquellen gespeichert sind. Der Zugriff auf Informix-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält eine Liste der Tasks, die Sie ausführen müssen, wenn Sie den Server mit zusammengeschlossenen Datenbanken konfigurieren, zeigt Beispiele zu den benötigten SQL-Anweisungen und bietet Informationen zur Optimierung und Fehlerbehebung beim Konfigurieren des Servers mit zusammengeschlossenen Datenbanken.

---

### Hinzufügen von Informix zu einem Server mit zusammengeschlossenen Datenbanken

#### Hinzufügen von Informix-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Informix-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken auf Informix-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

#### Voraussetzungen:

- Ein DB2-Server, der für den Zusammenschluss von Datenquellen konfiguriert ist.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.
- Die Software Informix Client SDK muss auf dem Server mit zusammengeschlossenen Datenbanken installiert und konfiguriert sein.
- Unter AIX-Servern mit zusammengeschlossenen Datenbanken: AIX Base Application Development Math Library. Durch Absetzen des AIX-Befehls **lslpp -l bos.adt.libm** können Sie ermitteln, ob diese Bibliothek (Library) installiert ist.

#### Vorgehensweise:

Gehen Sie wie folgt vor, um Informix-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Richten Sie die Informix-Clientkonfigurationsdatei ein und testen Sie diese Datei.
2. Legen Sie die Informix-Umgebungsvariablen fest.
3. Registrieren Sie den Wrapper.

4. Registrieren Sie die Serverdefinition.
5. Erstellen Sie die Benutzerzuordnungen.
6. Testen Sie die Verbindung zum Informix-Server.
7. Registrieren Sie Kurznamen für Informix-Tabellen, -Sichten und -Synonyme.

#### **Zugehörige Konzepte:**

- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

#### **Zugehörige Tasks:**

- „Überprüfung des Parameters FEDERATED“ auf Seite 50
- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Einrichten und Testen der Informix-Clientkonfigurationsdatei“ auf Seite 314
- „Registrieren des Informix-Wrappers“ auf Seite 319
- „Registrieren der Serverdefinitionen für eine Informix-Datenquelle“ auf Seite 321
- „Erstellen der Benutzerzuordnung für eine Informix-Datenquelle“ auf Seite 323
- „Testen der Verbindung zum Informix-Server“ auf Seite 325
- „Registrieren von Kurznamen für Informix-Tabellen, -Sichten und -Synonyme“ auf Seite 326
- „Optimierung und Fehlerbehebung der Konfiguration für Informix-Datenquellen“ auf Seite 328
- „Festlegen der Informix-Umgebungsvariablen“ auf Seite 315
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43

#### **Zugehörige Referenzen:**

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## **Einrichten und Testen der Informix-Clientkonfigurationsdatei**

Beim Hinzufügen von Informix-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken muss die Informix-Clientkonfigurationsdatei eingerichtet und getestet werden.

Die Clientkonfigurationsdatei wird zum Herstellen einer Verbindung zu Informix eingesetzt, wobei die Clientbibliotheken verwendet werden, die auf dem Server mit zusammengeschlossenen Datenbanken installiert sind. Diese Datei gibt die Speicherposition aller Informix-Datenbankserver und den Typ der Verbindung (Protokoll) für den Datenbankserver an.

Die standardmäßige Position der Clientkonfigurationsdatei ist abhängig von dem Betriebssystem des Servers mit zusammengeschlossenen Datenbanken.

- Unter UNIX-Betriebssystemen lautet die Standardposition und der Standardname der Datei `$INFORMIXDIR/etc/sqlhosts`. Die Datei `sqlhosts` wird mit der Informix-Client-SDK installiert.
- Unter Windows-Betriebssystemen ist die Registrierdatenbank `sqlhosts` standardmäßig auf dem lokalen Computer gespeichert.



Das Format von `sqlhosts` wird im Handbuch *Administrator's Guide for Informix Dynamic Server* beschrieben.

### Vorgehensweise:

Um die Informix-Clientkonfigurationsdatei einzurichten und zu testen, gehen Sie wie folgt vor:

1. Konfigurieren Sie Informix Client SDK.
  - Unter UNIX können Sie Informix Client SDK durch Editieren der Datei `sqlhosts` konfigurieren. Sie können die Datei `sqlhosts` auch von einem anderen System kopieren, auf dem Informix Connect oder Informix Client SDK installiert ist.
  - Unter Windows können Sie Informix Client SDK mit dem Informix-Dienstprogramm `Setnet32` konfigurieren. Das Dienstprogramm `Setnet32` richtet die Registrierdatenbank `sqlhosts` ein.
2. Überprüfen Sie die Speicherposition der Datei `sqlhosts` oder der Registrierdatenbank.
  - Unter UNIX-Betriebssystemen befindet sich die Datei `sqlhosts` im Verzeichnis `$INFORMIXDIR/etc/`.
  - Unter Windows-Betriebssystemen werden die `sqlhosts`-Informationen im folgenden Schlüssel in der Windows-Registrierdatenbank gespeichert:  
`HKEY_LOCAL_MACHINE\SOFTWARE\INFORMIX\SQLHOSTS`
3. Befindet sich die Datei `sqlhosts` oder die Registrierdatenbank nicht an der standardmäßigen Speicherposition, müssen Sie die Umgebungsvariable `INFORMIXSQLHOSTS` definieren.
  - Unter UNIX-Betriebssystemen: Setzen Sie die Umgebungsvariable `INFORMIXSQLHOSTS` auf den vollständig qualifizierten Namen der Datei `sqlhosts`.
  - Unter Windows-Betriebssystemen: Setzen Sie die Umgebungsvariable `INFORMIXSQLHOSTS` auf den Namen des Windows-Computers, auf dem die Registrierdatenbank gespeichert ist.
4. Testen Sie die Verbindung, um sicherzustellen, dass die Clientsoftware eine Verbindung zum Informix-Server herstellen kann. Wenn das Informix-Tool **`dbaccess`** auf dem Server mit zusammengeschlossenen Datenbanken vorhanden ist, verwenden Sie dieses Tool zum Testen der Verbindung. Führen Sie andernfalls das Informix-Demoprogramm aus, um die Clienteinrichtung zu testen.

Die nächste Task in dieser Tasksequenz ist das Festlegen der Umgebungsvariablen für Informix.

### Zugehörige Tasks:

- „Registrieren des Informix-Wrappers“ auf Seite 319
- „Optimierung und Fehlerbehebung der Konfiguration für Informix-Datenquellen“ auf Seite 328

## Festlegen der Informix-Umgebungsvariablen

Das Festlegen der Informix-Umgebungsvariablen ist ein Teil der umfangreicheren Task des Hinzufügens von Informix zu einem Server mit zusammengeschlossenen Datenbanken.



Bei der Installation von DB2 Information Integrator versucht der Installationsprozess, die Informix-Umgebungsvariablen in der Datei `db2dj.ini` festzulegen.

In folgenden Fällen werden die Umgebungsvariablen in der Datei `db2dj.ini` nicht festgelegt:

- Sie installieren die Informix-Client-Software, nachdem der DB2-Server mit zusammengeschlossenen Datenbanken konfiguriert wurde.
- Sie haben die Informix-Client-Software nicht installiert.

Gültige Umgebungsvariablen für Informix lauten wie folgt:

- `INFORMIXDIR`
- `INFORMIXSERVER`
- `INFORMIXSQLHOSTS` (optional)
- `CLIENT_LOCALE` (optional)
- `DB_LOCALE` (optional)
- `DBNLS` (optional)

Die optionalen Umgebungsvariablen müssen manuell festgelegt werden.

#### **Voraussetzungen:**

Ein Server mit zusammengeschlossenen Datenbanken, der für den Zugriff auf Ihre Datenquellen korrekt konfiguriert ist. Dies beinhaltet die Installation und Konfiguration von erforderlicher Software, z. B. der Client-Software der Datenquelle.

#### **Einschränkungen:**

Siehe hierzu den Abschnitt: Einschränkungen für die Datei `db2dj.ini`

#### **Vorgehensweise:**

Gehen Sie wie folgt vor, um die erforderlichen Umgebungsvariablen automatisch festzulegen:

1. Installieren und konfigurieren Sie die Client-Software auf dem DB2-Server mit zusammengeschlossenen Datenbanken (falls erforderlich).
2. Legen Sie die erforderlichen Umgebungsvariablen fest. Sie können die Umgebungsvariablen automatisch festlegen, indem Sie die Installation des DB2-Servers erneut ausführen und die Option für typische Installation angeben.

Gehen Sie wie folgt vor, um die Umgebungsvariablen manuell festzulegen:

1. Bearbeiten Sie die Datei `db2dj.ini`.
  - Auf unter Windows ausgeführten Servern mit zusammengeschlossenen Datenbanken befindet sich diese Datei im Verzeichnis `%DB2PATH%\cfg`.
  - Auf unter UNIX ausgeführten Servern mit zusammengeschlossenen Datenbanken befindet sich diese Datei im Verzeichnis `sql1lib/cfg`.

Die Datei `db2dj.ini` enthält Konfigurationsinformationen über die Informix-Client-Software, die auf Ihrem Server mit zusammengeschlossenen Datenbanken installiert ist. Ist die Datei nicht vorhanden, können Sie mit Hilfe eines beliebigen Texteditors eine neue Datei mit dem Namen `db2dj.ini` erstellen. In der Datei `db2dj.ini` müssen Sie den vollständig qualifizierten Pfad für die Variablen angeben, andernfalls treten Fehler auf.

2. Legen Sie die folgenden Umgebungsvariablen wie erforderlich fest:

### INFORMIXDIR

Setzen Sie die Umgebungsvariable INFORMIXDIR auf den Verzeichnispfad, in dem die Software von Informix Client SDK installiert ist. Beispiel:

Geben Sie folgenden Pfad auf Servern mit zusammengeschlossenen Datenbanken an, die unter Windows ausgeführt werden:

```
INFORMIXDIR=C:\informix\csdk
```

Geben Sie folgenden Pfad auf Servern mit zusammengeschlossenen Datenbanken an, die unter UNIX ausgeführt werden:

```
INFORMIXDIR=/informix/csdk
```

### INFORMIXSERVER

Diese Variable gibt den Namen des standardmäßigen Informix-Servers an. Diese Einstellung muss ein gültiger Eintrag in der Datei `sqlhosts` (UNIX) oder im `SQLHOSTS`-Registrierungsschlüssel (Windows) sein. Um einen Wert für INFORMIXSERVER abzurufen, lesen Sie die Datei `sqlhosts`. Wählen Sie einen der Werte für `dbservername` aus. Der `dbservername` ist der erste Wert in jedem Eintrag in `sqlhosts`. Beispiel:

```
INFORMIXSERVER=inf93
```

**Anforderung:** Obwohl der Informix-Wrapper nicht den Wert dieser Variable verwendet, erfordert der Informix-Client die Einstellung dieser Variable. Der Wrapper verwendet den Wert der Serveroption **NODE**, die den Informix-Datenbankserver angibt, auf den Sie zugreifen möchten.

### INFORMIXSQLHOSTS

Wenn Sie den Standardpfad für die Informix-Datei `sqlhosts` verwenden, müssen Sie diese Variable nicht festlegen. Bei Verwendung eines anderen Pfades für die Informix-Datei `sqlhosts` muss diese Variable jedoch auf den vollständigen Pfadnamen festgelegt werden, in dem sich die Informix-Datei `sqlhosts` befindet.

- Für Server mit zusammengeschlossenen Datenbanken, die unter UNIX ausgeführt werden, lautet der Standardpfad `$INFORMIXDIR/etc`.
- Wenn sich auf Windows-Servern mit zusammengeschlossenen Datenbanken der `SQLHOSTS`-Registrierungsschlüssel nicht auf dem lokalen Computer befindet, lautet der Name des Windows-Computers, der die Registrierdatenbank speichert, `INFORMIXSQLHOSTS`.

Nachfolgend ist ein Beispiel für UNIX aufgeführt, um diese Variable auf einen anderen Pfad festzulegen:

```
INFORMIXSQLHOSTS=/informix/csdk/etc/my_sqlhosts
```

3. Um sicherzustellen, dass die Umgebungsvariablen für den Server mit zusammengeschlossenen Datenbanken gesetzt sind, beenden Sie das DB2-Exemplar, und starten Sie es erneut. Geben Sie die folgenden Befehle aus, um das DB2-Exemplar erneut zu starten:

```
db2stop  
db2start
```

## Einrichten der Codepagekonvertierung für Informix:

Zur Informix-Codepagekonvertierung können Sie folgende optionale Umgebungsvariablen festlegen:

- CLIENT\_LOCALE
- DB\_LOCALE
- DBNLS

Jedes Mal, wenn der Informix-Wrapper eine Verbindung zu einer Informix-Datenquelle herstellt, ermittelt der Wrapper den für diese Verbindung zu verwendenden Codepagewert. Ist die Informix-Umgebungsvariable CLIENT\_LOCALE auf dem Server mit zusammengeschlossenen Datenbanken in der Datei `db2dj.ini` festgelegt, verwendet der Wrapper den Wert in der Datei `db2dj.ini`. Sie können die Liste der gültigen Informix-Ländereinstellungen durch Ausgabe des Befehls `glfiles` auf dem Informix-Server abrufen. Weitere Informationen zu Codepagekonvertierungen finden Sie in der Dokumentation *Informix Guide to GLS Functionality*.

Die Umgebungsvariablen für die Informix-Codepage lauten wie folgt:

### CLIENT\_LOCALE

Setzen Sie die Umgebungsvariable CLIENT\_LOCALE auf die Informix-Ländereinstellung, die Sie verwenden möchten. Ist CLIENT\_LOCALE nicht festgelegt, ermittelt der Wrapper die Codepage und das Gebiet der zusammengeschlossenen Datenbank. Die Variable CLIENT\_LOCALE wird vom Wrapper dann auf diejenige Informix-Ländereinstellung mit der höchsten Übereinstimmung gesetzt. Gibt es keine übereinstimmende Informix-Ländereinstellung, setzt der Wrapper die Variable CLIENT\_LOCALE für UNIX-Systeme auf die Ländereinstellung `en_us.8859-1` und für Windows-Systeme auf die Ländereinstellung `en_us.CP1252`. Mit Hilfe des Informix-Befehls `glfiles` können Sie eine Liste der Namen für Ländereinstellungen anzeigen lassen.

```
CLIENT_LOCALE=ländereinstellungswert_für_Informix-client
```

### DB\_LOCALE

Legen Sie diese Umgebungsvariable fest, falls die Informix-Datenbank eine andere Codepage als die Ländereinstellung des Clients verwendet und Informix Konvertierungen zwischen diesen beiden Codepages ausführen soll. Setzen Sie die Informix-Umgebungsvariable DB\_LOCALE auf den Namen für die Ländereinstellung der Informix-Datenbank. Beispiel:

```
DB_LOCALE=ländereinstellungswert_für_Informix-db
```

### DBNLS

Um Informix bestätigen zu lassen, dass die Einstellung DB\_LOCALE mit der tatsächlichen Ländereinstellung der Informix-Datenbank übereinstimmt, setzen Sie diese Informix-Umgebungsvariable auf 1.

```
DBNLS=1
```

### Umgebungsvariablen für Codepages - Beispiel:

Angenommen, die Informix-Datenbank verwendet eine andere Codepage als die Ländereinstellung Ihres Clients, und Informix soll Konvertierungen zwischen diesen beiden Codepages ausführen. Dann müssen Sie die Informix-Umgebungsvariable DB\_LOCALE auf den Namen der Ländereinstellung für die Informix-Datenbank setzen. Legen Sie diese Variable auf dem Server mit zusammengeschlossenen Datenbanken in der Datei `'db2dj.ini'` fest.

Um Informix bestätigen zu lassen, dass die Einstellung DB\_LOCALE mit der tatsächlichen Ländereinstellung der Informix-Datenbank übereinstimmt, setzen Sie die Informix-Umgebungsvariable DBNLS auf 1. Diese Variable wird auf dem Server mit zusammengeschlossenen Datenbanken in der Datei 'db2dj.ini' festgelegt.

Wenn Sie auf eine Datenquelle mit Daten der chinesischen Codepage GB 18030 zugreifen, muss Ihre zusammengeschlossene Datenbank die Codepage UTF-8 anwenden. Der Informix-Wrapper legt Informix-Umgebungsvariablen wie folgt fest:

```
CLIENT_LOCALE=zh_cn.UTF8
GL_USEGLU=1
```

Sie müssen die folgende Einstellung zu Ihrer Datei db2dj.ini hinzufügen, damit der Informix-Client die GB 18030-Daten korrekt in Unicode umsetzt:

```
DB_LOCALE=zh_cn.GB18030-2000
```

Die nächste Task in dieser Tasksequenz ist das Registrieren des Informix-Wrappers.

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für Informix-Tabellen, -Sichten und -Synonyme“ auf Seite 326

#### Zugehörige Referenzen:

- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68
- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68

## Registrieren des Informix-Wrappers

Die Registrierung des Informix-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von Informix-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf Informix-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

#### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER aus, und geben Sie den Standardnamen für den Wrapper an.

Beispiel:

```
CREATE WRAPPER INFORMIX
```

**Empfehlung:** Verwenden Sie den Standardwrappernamen INFORMIX. Wenn Sie den Wrapper mit dem Standardnamen registrieren, übernimmt der Server mit zusammengeschlossenen Datenbanken automatisch den standardmäßigen Bibliotheksnamen, der diesem Wrapper-Namen zugeordnet ist.

Wenn der Wrapper-Name zu Konflikten mit einem bestehenden Wrapper-Namen in der zusammengeschlossenen Datenbank führt, können Sie den standardmäßigen Wrapper-Namen durch einen Namen Ihrer Wahl ersetzen. Wenn Sie einen Namen verwenden, der sich von einem der Standardnamen unterscheidet, müssen Sie in der Anweisung CREATE WRAPPER den Parameter LIBRARY mit angeben.

Um beispielsweise einen Wrapper mit dem Namen `inf_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER inf_wrapper LIBRARY 'libdb2informix.a'
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung `CREATE WRAPPER` finden Sie in der Liste der Bibliotheksdateien von Informix-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den Informix-Wrapper.

**Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine Informix-Datenquelle“ auf Seite 321

**Zugehörige Referenzen:**

- „Bibliotheksdateien für den Informix-Wrapper“ auf Seite 320
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien für den Informix-Wrapper

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den Informix-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, werden 3 Bibliotheksdateien zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden folgende Wrapperbibliotheksdateien zum Verzeichnispfad hinzugefügt: `libdb2informix.a`, `libdb2informixF.a` und `libdb2informixU.a`.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 76. Positionen und Dateinamen von Informix-Wrapperbibliotheken*

Betriebssystem	Verzeichnispfad	Bibliotheksdateiname
AIX	/usr/opt/db2_08_01/lib/	libdb2informix.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2informix.sl
Linux	/opt/IBM/db2/V8.1/lib	libdb2informix.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2informix.so
Windows	%DB2PATH%\bin	db2informix.dll

`%DB2PATH%` ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist `C:\Program Files\IBM\SQLLIB`.

**Zugehörige Tasks:**

- „Registrieren des Informix-Wrappers“ auf Seite 319

## Registrieren der Serverdefinitionen für eine Informix-Datenquelle

Beim Hinzufügen von Informix-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen die Serverdefinitionen für eine Informix-Datenquelle registriert werden.

In der zusammengeschlossenen Datenbank müssen Sie jeden Informix-Server definieren, auf den Sie zugreifen möchten.

### Vorgehensweise:

Sie können eine Serverdefinition von der DB2-Steuerzentrale aus oder über die DB2-Befehlszeile registrieren:

- Um diese Task von der DB2-Steuerzentrale aus auszuführen, verwenden Sie den Assistenten für zusammengeschlossene Objekte oder klicken Sie mit der rechten Maustaste den Ordner **Serverdefinitionen** an, und klicken Sie anschließend **Erstellen** an. Das Tool zum Aufspüren (Discover-Tool) ruft die Knotennamen für die Informix-Server ab. Sie müssen die Informationen für die Serveroption DBNAME angeben, um die Serverdefinition zu registrieren.
- Um diese Task von der DB2-Befehlszeile aus auszuführen, verwenden Sie die Anweisung CREATE SERVER:
  1. Suchen Sie den Knotennamen in der Informix-Datei sqlhosts oder der Informix-Registrierdatenbank.

#### Beispieldatei sqlhosts:

```
inf724 onsoctcp anaconda inmx724
inf731 onsoctcp boa ifmx731
inf92 onsoctcp python ifmx92
```

Der erste Wert in den einzelnen Zeilen ist der Parameter *knotenname* wie beispielsweise inf724.

Der zweite Wert in den einzelnen Zeilen ist der Parameter *netztyp* bzw. der Verbindungstyp. In diesem Beispiel gibt onsoctcp an, dass es sich um eine TCP/IP-Verbindung handelt.

Der dritte Wert in den einzelnen Zeilen ist der Hostname, wie z. B. 'anaconda', 'boa' und 'python'.

Der vierte Wert in den einzelnen Zeilen ist der Servicenamen wie beispielsweise inmx724. Das Feld für den Servicenamen hängt vom Parameter *netztyp* (der zweite aufgelistete Wert) ab.

Obwohl der Parameter *knotenname* in der SQL-Anweisung CREATE SERVER als Option angegeben wird, ist er für Informix-Datenquellen erforderlich.

Weitere Informationen zum Format dieser Datei und zur Bedeutung dieser Felder finden Sie im Informix-Handbuch *Administrators Guide for Informix Dynamic Server*.

2. Setzen Sie die Anweisung CREATE SERVER ab.

Beispiel:

```
CREATE SERVER servername TYPE informix
VERSION 9 WRAPPER INFORMIX
OPTIONS (NODE 'knotenname', DBNAME 'db-name');
```

Verwenden Sie nach dem Registrieren der Serverdefinition die Anweisung ALTER SERVER, um Serveroptionen hinzuzufügen oder zu löschen.

Die nächste Task in dieser Tasksequenz ist das Erstellen der Benutzerzuordnungen für eine Informix-Datenquelle.

**Zugehörige Tasks:**

- „Erstellen der Benutzerzuordnung für eine Informix-Datenquelle“ auf Seite 323

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641
- „Anweisung CREATE SERVER - Beispiele für den Informix-Wrapper“ auf Seite 322

## Anweisung CREATE SERVER - Beispiele für den Informix-Wrapper

In diesem Abschnitt sind mehrere Beispiele für die Verwendung der Anweisung CREATE SERVER zur Registrierung von Serverdefinitionen für den Informix-Wrapper enthalten. Der Abschnitt enthält ein vollständiges Beispiel für die Registrierung einer Serverdefinition mit den erforderlichen Parametern sowie ein Beispiel mit zusätzlichen Serveroptionen.

**Vollständiges Beispiel:**

Das folgende Beispiel zeigt, wie eine Serverdefinition für einen Informix-Wrapper mit Hilfe der Anweisung CREATE SERVER registriert wird:

```
CREATE SERVER asia TYPE informix VERSION 9 WRAPPER INFORMIX
OPTIONS (NODE 'abc', DBNAME 'sales', IUD_APP_SVPT_ENFORCE 'N')
```

*asia* Ein Name, den Sie dem Informix-Datenbankserver zuordnen. Dieser Name muss eindeutig sein. Doppelte Servernamen sind nicht zulässig.

**TYPE** *informix*

Gibt den Typ des Datenquellenservers an, für den Sie den Zugriff konfigurieren. Für den Informix-Wrapper muss der Servertyp *informix* lauten.

**VERSION** 9

Die Version des Informix-Datenbankservers, auf den Sie zugreifen möchten. Die Informix-Versionen 7, 8 und 9 werden unterstützt.

**WRAPPER** *INFORMIX*

Der Name, den Sie in der Anweisung CREATE WRAPPER angeben haben.

**NODE** *'abc'*

Der Name des Knotens, in dem sich der Informix-Datenbankserver befindet. Den Knotennamen finden Sie in der Datei *sqlhosts*. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

Obwohl der Knotenname in der Anweisung CREATE SERVER als Option angegeben wird, ist er für Informix-Datenquellen erforderlich.

**DBNAME** *'sales'*

Der Name der Informix-Datenbank, auf die Sie zugreifen möchten. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

Obwohl der Datenbankname in der Anweisung CREATE SERVER als Option angegeben wird, ist er für Informix-Datenquellen erforderlich.



### IUD\_APP\_SVPT\_ENFORCE 'N'

Gibt an, ob das DB2-System zusammengeschlüssener Datenbanken das Finden oder Erstellen von Anweisungen für Anwendungssicherungspunkte durchsetzen soll. Informix unterstützt keine Anweisungen für Anwendungssicherungspunkte. Bei der Einstellung 'N' macht der Server mit zusammengeschlüssenen Datenbanken beim Auftreten eines Fehlers Transaktionen nicht rückgängig. Die Fehlerbehebung muss von Ihrer Anwendung übernommen werden.

Die Serveroption IUD\_APP\_SVPT\_ENFORCE muss auf 'N' eingestellt werden, um das Replizieren für oder aus Informix-Datenquellen zu aktivieren.

### Beispiel mit Serveroptionen:

Beim Erstellen der Serverdefinition können Sie in der Anweisung CREATE SERVER zusätzliche Serveroptionen angeben. Diese Serveroptionen können allgemein oder Informix-spezifisch sein.

Das folgende Beispiel zeigt eine Informix-Serverdefinition mit zusätzlichen Serveroptionen:

```
CREATE SERVER asia TYPE informix VERSION 9 WRAPPER INFORMIX
      OPTIONS (NODE 'abc', DBNAME 'sales', FOLD_ID 'N', FOLD_PW 'N')
```

Wenn der Server mit zusammengeschlüssenen Datenbanken eine Verbindung zu einer Datenquelle herstellt, versucht er dies unter Verwendung aller möglichen Kombinationen aus Groß- und Kleinbuchstaben für die Benutzer-ID und das Kennwort sowie der aktuellen Groß-/Kleinschreibung. Der Server mit zusammengeschlüssenen Datenbanken unternimmt gegebenenfalls bis zu neun Versuche zum Herstellen der Verbindung zum Datenquellenserver, bevor dieser Vorgang erfolgreich verläuft. Durch diese Versuche kann es zu einer Reduzierung der Verarbeitungsgeschwindigkeit für Verbindungen und zum Sperren der Benutzer-ID kommen. Das Sperren kann durch die Angabe von Werten für die Serveroptionen FOLD\_ID und FOLD\_PW vermieden werden.

Beispielsweise können Sie die Serveroptionen FOLD\_ID und FOLD\_PW auf 'N' setzen (Benutzer-ID oder Kennwort nicht umsetzen). Wenn Sie diese Einstellungen verwenden, müssen Sie die Benutzer-ID und das Kennwort in der richtigen Schreibung angeben. Werden diese Optionen auf 'N' gesetzt, hat dies den Vorteil, dass bei Angabe einer ungültigen Benutzer-ID oder eines ungültigen Kennworts der Wrapper die verschiedenen Kombinationen aus Groß- und Kleinschreibung nicht weiterhin anzuwenden versucht. Diese beiden Serveroptionen reduzieren die Wahrscheinlichkeit, dass die maximal zulässige Anzahl an fehlgeschlagenen Anmeldeversuchen überschritten und die ID gesperrt wird.

### Zugehörige Tasks:

- „Registrieren der Serverdefinitionen für eine Informix-Datenquelle“ auf Seite 321

### Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

## Erstellen der Benutzerzuordnung für eine Informix-Datenquelle

Beim Hinzufügen von Informix-Datenquellen zu Servern mit zusammengeschlüssenen Datenbanken muss die Benutzerzuordnung für eine Informix-Datenquelle erstellt werden.



Wird versucht, auf einen Informix-Server zuzugreifen, stellt der Server mit zusammengeschlossenen Datenbanken eine Verbindung zur Datenquelle her. Hierbei verwendet er eine Benutzer-ID und ein Kennwort, die für die betreffende Datenquelle gültig sind. Sie müssen eine Zuordnung (eine Benutzerzuordnung) zwischen jeder Benutzer-ID sowie jedem Kennwort des Servers mit zusammengeschlossenen Datenbanken und der entsprechenden Benutzer-ID und dem entsprechenden Kennwort der Datenquelle definieren. Erstellen Sie eine Benutzerzuordnung für jede Benutzer-ID, über die auf das System zusammengeschlossener Datenbanken zugegriffen wird, um verteilte Anforderungen an die Informix-Datenquelle zu senden.

#### Vorgehensweise:

Um der Benutzer-ID und dem Kennwort des Informix-Servers eine lokale Benutzer-ID zuzuordnen, setzen Sie die Anweisung CREATE USER MAPPING ab.

Beispiel:

```
CREATE USER MAPPING FOR USERID SERVER INFORMIXSERVER
    OPTIONS (REMOTE_AUTHID 'ferne-id', REMOTE_PASSWORD 'fernes_kennwort')
```

Die nächste Task in dieser Tasksequenz ist das Testen der Verbindung zum Informix-Server.

#### Zugehörige Tasks:

- „Testen der Verbindung zum Informix-Server“ auf Seite 325

#### Zugehörige Referenzen:

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE USER MAPPING - Beispiel für den Informix-Wrapper“ auf Seite 324

## Anweisung CREATE USER MAPPING - Beispiel für den Informix-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE USER MAPPING verwendet wird, um die Benutzer-ID eines Servers mit zusammengeschlossenen Datenbanken einer Benutzer-ID und einem Kennwort eines Informix-Servers zuzuordnen. Dieser Abschnitt enthält ein vollständiges Beispiel mit den erforderlichen Parametern sowie ein Beispiel, das zeigt, wie das DB2-Sonderregister USER mit der Anweisung CREATE USER MAPPING verwendet wird.

#### Vollständiges Beispiel:

Das folgende Beispiel zeigt, wie die Benutzer-ID (VINCENT) eines Servers mit zusammengeschlossenen Datenbanken der Benutzer-ID und dem Kennwort ('vinnie' und 'close2call') eines Informix-Servers zugeordnet wird:

```
CREATE USER MAPPING FOR VINCENT SERVER asia
    OPTIONS (REMOTE_AUTHID 'vinnie', REMOTE_PASSWORD 'close2call')
```

**VINCENT**

Gibt die lokale Benutzer-ID an, die Sie einer Benutzer-ID zuordnen, die auf einem Informix-Server definiert wurde.

**SERVER asia**

Gibt den Namen des Informix-Servers an, den Sie in der Anweisung CREATE SERVER registriert haben.

**REMOTE\_AUTHID 'vinnie'**

Gibt die Benutzer-ID auf dem Informix-Datenbankserver an, zu der Sie *VINCENT* zuordnen. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption *FOLD\_ID* in der Anweisung *CREATE SERVER* nicht auf 'U' oder 'L' setzen.

**REMOTE\_PASSWORD 'close2call'**

Gibt das Kennwort an, das 'vinnie' zugeordnet ist. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption *FOLD\_PW* in der Anweisung *CREATE SERVER* nicht auf 'U' oder 'L' setzen.

**Beispiel für das Sonderregister:**

Das folgende Beispiel zeigt eine Anweisung *CREATE USER MAPPING*, die das Sonderregister *USER* enthält:

```
CREATE USER MAPPING FOR USER SERVER asia
  OPTIONS (REMOTE_AUTHID 'vinnie', REMOTE_PASSWORD 'close2call')
```

Sie können das DB2-Sonderregister *USER* verwenden, um die Berechtigungs-ID der Person, die die Anweisung *CREATE USER MAPPING* absetzt, zur Berechtigungs-ID der Datenquelle zuzuordnen, die in der Benutzeroption *REMOTE\_AUTHID* angegeben wurde.

**Zugehörige Tasks:**

- „Erstellen der Benutzerzuordnung für eine Informix-Datenquelle“ auf Seite 323

**Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

## Testen der Verbindung zum Informix-Server

Beim Hinzufügen von Informix-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken muss die Verbindung zum Informix-Server getestet werden.

Sie können die Verbindung zum Informix-Server testen, indem Sie die Serverdefinition und Benutzerzuordnungen verwenden, die Sie definiert haben.

**Vorgehensweise:**

Um die Verbindung zu testen, gehen Sie wie folgt vor:

1. Öffnen Sie eine Durchgriffssitzung, um eine SQL-Anweisung *SELECT* für die Informix-Systemtabellen abzusetzen.

Beispiel:

```
SET PASSTHRU servername
SELECT count(*) FROM informix.systables
SET PASSTHRU RESET
```

Wenn die SQL-Anweisung *SELECT* eine Anzahl zurückgibt, sind Ihre Serverdefinition und Benutzerzuordnung ordnungsgemäß eingerichtet.

2. Wenn die SQL-Anweisung *SELECT* einen Fehler zurückgibt, müssen Sie möglicherweise folgende Aktionen ausführen:
  - Den Informix-Server prüfen, um sicherzustellen, dass er für ankommende Verbindungsanforderungen konfiguriert wurde.

- Ihre Benutzerzuordnung prüfen, um sicherzustellen, dass die Einstellungen für die Optionen REMOTE\_AUTHID und REMOTE\_PASSWORD für Verbindungen zum Informix-Server gültig sind. Ändern Sie gegebenenfalls die Benutzerzuordnung oder erstellen Sie gegebenenfalls eine andere Benutzerzuordnung.
- Die Software 'Informix Client SDK auf dem DB2-Server mit zusammengeschlossenen Datenbanken prüfen, um sicherzustellen, dass sie für das Herstellen einer Verbindung zum Informix-Server ordnungsgemäß installiert und konfiguriert wurde.
- Die Einstellungen Ihrer DB2-Variablen für zusammengeschlossene Datenbanken prüfen, um sicherzustellen, dass sie für das Arbeiten mit dem Informix-Server korrekt definiert sind. Zu diesen Variablen gehören die Systemumgebungsvariablen, die db2dj.ini-Variablen und die Variable der DB2-Profilregistrierdatenbank (db2set).
- Ihre Serverdefinition prüfen. Gegebenenfalls müssen Sie diese Definition löschen und erneut erstellen.

Die nächste Task in dieser Tasksequenz ist das Registrieren von Kurznamen für Informix-Tabellen, -Sichten und -Synonyme.

#### **Zugehörige Tasks:**

- „Hinzufügen von Informix-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken“ auf Seite 313
- „Registrieren von Kurznamen für Informix-Tabellen, -Sichten und -Synonyme“ auf Seite 326
- „Festlegen der Informix-Umgebungsvariablen“ auf Seite 315

#### **Zugehörige Referenzen:**

- „ALTER USER MAPPING statement“ in *SQL Reference, Volume 2*

## **Registrieren von Kurznamen für Informix-Tabellen, -Sichten und -Synonyme**

Beim Hinzufügen von Informix-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen Kurznamen für Informix-Tabellen, -Sichten und -Synonyme registriert werden.

Registrieren Sie für jeden Informix-Server, den Sie definieren, jeweils einen Kurznamen für alle Tabellen, Sichten oder Synonyme, auf die Sie zugreifen wollen. Verwenden Sie diese Kurznamen anstelle der Namen der Datenquellenobjekte, wenn Sie die Informix-Server abfragen.

Die zusammengeschlossene Datenbank verwendet Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Diese Statistiken werden gesammelt, wenn Sie unter Verwendung der Anweisung CREATE NICKNAME einen Kurznamen für ein Datenquellenobjekt registrieren. Die zusammengeschlossene Datenbank stellt das Vorhandensein des Objekts in der Datenquelle sicher und versucht dann, bestehende statistische Daten zur Datenquelle zu sammeln. Informationen, die für das Optimierungsprogramm nützlich sind, werden aus den Katalogen der Datenquelle gelesen und in den globalen Katalog auf dem Server mit zusammengeschlossenen Datenbanken aufgenommen. Da einige oder alle der Kataloginformationen zu Datenquellen vom Optimierungsprogramm verwendet werden können, sollten Sie die Statistiken (unter Verwendung des Daten-

quellenbefehls, der dem DB2-Befehl **RUNSTATS** entspricht) in der Datenquelle aktualisieren, bevor Sie einen Kurznamen registrieren.

#### Vorgehensweise:

Um einen Kurznamen zu registrieren, setzen Sie die Anweisung **CREATE NICKNAME** ab.

Beispiel:

```
CREATE NICKNAME informix-name FOR INFOSERVER."remote_schema"."remote.table"
```

Kurznamen können bis zu 128 Zeichen lang sein.

Wiederholen Sie diesen Schritt jeweils für alle Informix-Tabellen, -Sichten oder Synonyme, für die Sie einen Kurznamen erstellen möchten.

Wenn Sie den Kurznamen erstellen, verwendet DB2 die Verbindung dazu, den Datenquellenkatalog abzufragen. Diese Abfrage testet Ihre Verbindung zur Datenquelle unter Verwendung des Kurznamens. Wenn die Verbindung nicht funktioniert, empfangen Sie eine Fehlermeldung.

#### Zugehörige Referenzen:

- „RUNSTATS Command“ in *Command Reference*
- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den Informix-Wrapper“ auf Seite 327

## Anweisung CREATE NICKNAME - Beispiele für den Informix-Wrapper

Dieser Abschnitt enthält ein Beispiel, das zeigt, wie die Anweisung **CREATE NICKNAME** verwendet wird, um einen Kurznamen für Informix-Tabellen, -Sichten oder -Synonyme zu registrieren, auf die Sie zugreifen möchten.

Dieses Beispiel zeigt, wie ein fernes Objekt für den Informix-Server angegeben wird, unter dem der Kurzname zugeordnet ist:

```
CREATE NICKNAME JPSALES FOR asia."salesdata"."japan"
```

*JPSALES*

Ein eindeutiger Kurzname, der zur Kennzeichnung der Informix-Tabelle oder -Sicht bzw. des Informix-Synonyms verwendet wird.

Anmerkung: Der Kurzname ist ein zweiteiliger Name - das Schema und der Kurzname. Wenn Sie bei der Registrierung des Kurznamens das Schema auslassen, wird die Berechtigungs-ID des Benutzers, der den Kurznamen registriert, zum Schema des Kurznamens.

*asia."salesdata"."japan"*

Eine dreiteilige Kennung für das ferne Objekt.

- *asia* ist der Name, den Sie dem Informix-Datenbankserver in der Anweisung **CREATE SERVER** zugeordnet haben.
- *salesdata* ist der Name des fernen Schemas, zu dem die Tabelle oder Sicht bzw. das Synonym gehört.
- *japan* ist der Name der fernen Tabelle oder Sicht bzw. des fernen Synonyms, auf die/das Sie zugreifen möchten.

Der Server mit zusammengeschlossenen Datenbanken setzt die Namen der Informix-Schemata und -Tabellen in Großbuchstaben um, sofern Sie die Namen nicht in Anführungszeichen setzen.

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Informix-Tabellen, -Sichten und -Synonyme“ auf Seite 326

**Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

---

## Optimierung und Fehlerbehebung der Konfiguration für Informix-Datenquellen

Nachdem Sie die Konfiguration für Informix-Datenquellen eingerichtet haben, müssen Sie diese Konfiguration möglicherweise ändern, um die Leistung zu verbessern.

### Verbessern der Leistung durch Definieren der Serveroptionen FOLD\_ID und FOLD\_PW

Wenn der Server mit zusammengeschlossenen Datenbanken eine Verbindung zu einer Datenquelle herstellt, versucht er dies unter Verwendung aller möglichen Kombinationen aus Groß- und Kleinbuchstaben für die Benutzer-ID und das Kennwort. Der Server unternimmt gegebenenfalls bis zu neun Versuche zum Herstellen der Verbindung zum Datenquellenserver, bevor dieser Vorgang erfolgreich verläuft. Durch diese Versuche kann es zu einer Reduzierung der Verarbeitungsgeschwindigkeit für Verbindungen und zum Sperren der Benutzer-ID kommen.

**Vorgehensweise:**

Um die Leistung zu verbessern, können Sie mit Hilfe der Anweisung ALTER SERVER OPTION Werte für die Serveroptionen FOLD\_ID und FOLD\_PW angeben.

- Wenn sämtliche Ihrer Informix-Benutzer-IDs und -Kennwörter in Kleinschreibung sind, kann Ihre Verbindungszeit unter Umständen verbessert werden, indem Sie die Serveroptionen FOLD\_ID und FOLD\_PW auf den Wert 'L' setzen. Beispiel:

```
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_ID 'L');
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_PW 'L');
```

- Der Server mit zusammengeschlossenen Datenbanken versucht alle Kombinationen der Werte in Groß- und Kleinschreibung für die Benutzer-ID und das Kennwort. Sie können die Wahrscheinlichkeit reduzieren, dass die maximal zulässige Anzahl an fehlgeschlagenen Anmeldeversuchen überschritten wird, indem Sie diese Optionen auf 'N' (Benutzer-ID und Kennwort nicht umsetzen) einstellen. Wenn Sie diese Einstellungen verwenden, müssen Sie die Benutzer-ID und das Kennwort stets in der richtigen Schreibung angeben. Bei Angabe einer ungültigen Benutzer-ID und eines ungültigen Kennworts versucht der Wrapper nicht, die verschiedenen Kombinationen weiterhin anzuwenden. Beispiel:

```
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_ID 'N');
ALTER SERVER TYPE INFORMIX
  OPTIONS (ADD FOLD_PW 'N');
```

**Zugehörige Tasks:**

- „Hinzufügen von Informix-Datenquellen zu Servern mit zusammengesetzten Datenbanken“ auf Seite 313

**Zugehörige Referenzen:**

- „db2set - DB2 Profile Registry Command“ in *Command Reference*
- „ALTER SERVER statement“ in *SQL Reference, Volume 2*



---

## Kapitel 16. Konfigurieren des Zugriffs auf Microsoft SQL Server-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Microsoft SQL Server-Datenquellen gespeichert sind. Der Zugriff auf Microsoft SQL Server-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält eine Liste der Tasks, die Sie ausführen müssen, wenn Sie den Server mit zusammengeschlossenen Datenbanken konfigurieren, zeigt Beispiele zu den benötigten SQL-Anweisungen und bietet Informationen zur Optimierung und Fehlerbehebung beim Konfigurieren des Servers mit zusammengeschlossenen Datenbanken.

---

### Hinzufügen von Microsoft SQL Server zu einem Server mit zusammengeschlossenen Datenbanken

#### Hinzufügen von Microsoft SQL Server-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Microsoft SQL Server-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken auf Microsoft SQL Server-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

#### Voraussetzungen:

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.
- Der ODBC-Treiber muss auf dem Server mit zusammengeschlossenen Datenbanken installiert und konfiguriert sein. Unterstützte Treiber sind der Microsoft ODBC-Treiber (Windows) und der Treiber von DataDirect Technologies Connect für ODBC (UNIX).

#### Vorgehensweise:

Führen Sie die folgenden Schritte aus, um Microsoft SQL Server-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Bereiten Sie den Server mit zusammengeschlossenen Datenbanken und die zusammengeschlossene Datenbank vor.



- Unter Windows: Vergewissern Sie sich, dass der ODBC-System-DSN korrekt eingerichtet ist, und testen Sie die Verbindung zum fernen Server mit Microsoft SQL Server.
  - Unter UNIX-Systemen: Aktualisieren oder Erstellen Sie die Datei `odbc.ini` und testen Sie die Verbindung zum fernen Server mit Microsoft SQL Server.
2. Setzen Sie die Umgebungsvariablen für den Microsoft SQL Server-Wrapper.
  3. Registrieren Sie den Wrapper.
  4. Registrieren Sie die Serverdefinition.
  5. Erstellen Sie die Benutzerzuordnungen.
  6. Testen Sie die Verbindung zum fernen Server mit Microsoft SQL Server.
  7. Registrieren Sie Kurznamen für Microsoft SQL Server-Tabellen und -Sichten.

#### **Zugehörige Konzepte:**

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

#### **Zugehörige Tasks:**

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Vorbereiten des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf Microsoft SQL Server-Datenquellen“ auf Seite 332
- „Registrieren des Microsoft SQL Server-Wrappers“ auf Seite 336
- „Registrieren der Serverdefinitionen für eine Microsoft SQL Server-Datenquelle“ auf Seite 338
- „Erstellen einer Benutzerzuordnung für eine Microsoft SQL Server-Datenquelle“ auf Seite 341
- „Testen der Verbindung zum fernen Server mit Microsoft SQL Server“ auf Seite 342
- „Registrieren von Kurznamen für Microsoft SQL Server-Tabellen und Sichten“ auf Seite 344
- „Optimieren der Konfiguration für Microsoft SQL Server-Datenquellen und Beheben von Konfigurationsfehlern“ auf Seite 345
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43

#### **Zugehörige Referenzen:**

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## **Vorbereiten des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf Microsoft SQL Server-Datenquellen**

Die Vorbereitung des Servers mit zusammengeschlossenen Datenbanken und der zusammengeschlossenen Datenbank für den Zugriff auf Microsoft SQL Server-Datenquellen ist Teil der umfangreicheren Task des Hinzufügens von Microsoft SQL Server-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken.

Die auszuführenden Schritte zur Vorbereitung des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf Microsoft SQL Server-Datenquellen sind abhängig von dem Betriebssystem, das auf Ihrem Server mit zusammengeschlossenen Datenbanken ausgeführt wird.

### Vorgehensweise:

Führen Sie folgende Schritte zur Vorbereitung des Servers mit zusammengeschlossenen Datenbanken aus:

#### Unter Windows::

1. Stellen Sie sicher, dass der ODBC-System-DSN für das Herstellen einer Verbindung zur Microsoft SQL Server-Datenquelle gesetzt wurde, indem Sie diese Einstellung in der Systemsteuerung überprüfen. Suchen Sie den vorhandenen Eintrag für den fernen Server von Microsoft SQL Server, oder erstellen Sie einen neuen Eintrag. Der Eintrag ist der Wert, den Sie für die Serveroption NODE verwenden, wenn Sie den Server in der zusammengeschlossenen Datenbank registrieren.
2. Wählen Sie im Fenster 'ODBC-Datenquellen-Administrator' die Option **Konfigurieren** aus, um die Verbindung vom ODBC-System-DSN zur Microsoft SQL Server-Datenquelle zu testen. Alternativ können Sie die Verbindung unter Verwendung des Microsoft SQL Server-Abfragetools testen.

#### Unter UNIX::

1. Stellen Sie sicher, dass die Datei `odbc.ini` auf dem Server mit zusammengeschlossenen Datenbanken aktualisiert (oder bei Bedarf erstellt) wurde.  
**Empfehlung:** Stellen Sie die Datei `odbc.ini` oder eine Kopie dieser Datei in das Ausgangsverzeichnis des DB2-Exemplareigners.
2. Stellen Sie sicher, dass der Pfad zur Datei `odbc.ini` in der Umgebungsvariablen `ODBCINI` enthalten ist. Setzen Sie den folgenden Befehl über eine Eingabeaufforderung des Betriebssystems ab:

```
export ODBCINI=$HOME/.odbc.ini
```

3. Erstellen Sie die entsprechenden symbolischen Verbindungen (Links):

- Unter HP-UX müssen Sie folgende symbolische Verbindung erstellen:

```
ln -s $DJX_ODBC_LIBRARY_PATH/libodbcinst.sl /usr/exe/libodbcinst.sl
```

Wenn Sie den Treiber von DataDirect Technologies Connect für ODBC 4.2 verwenden, müssen Sie zusätzlich folgende symbolische Verbindung erstellen:

```
ln -s $DJX_ODBC_LIBRARY_PATH/libivicu19.sl /ivicu/exe/libivicu19.sl
```

- Unter Linux müssen Sie folgende symbolische Verbindungen erstellen:

```
ln -s $DJX_ODBC_LIBRARY_PATH/./locale /usr/local/locale
```

```
ln -s $DJX_ODBC_LIBRARY_PATH/libodbcinst.so /usr/lib/libodbcinst.so
```

Wenn Sie den Treiber von DataDirect Technologies Connect für ODBC 4.2 verwenden, müssen Sie zusätzlich folgende symbolische Verbindung erstellen:

```
ln -s $DJX_ODBC_LIBRARY_PATH/libivicu19.so /usr/lib/libivicu19.so
```

- Unter Solaris müssen Sie folgende symbolische Verbindung erstellen:

```
ln -s $DJX_ODBC_LIBRARY_PATH/./locale $HOME/sql/lib/locale
```

`$HOME` ist das Ausgangsverzeichnis des DB2-Exemplareigners.

4. Testen Sie die Verbindung vom Server mit zusammengeschlossenen Datenbanken zur Microsoft SQL Server-Datenquelle mit Hilfe des ODBC-Tools **demodbc** von DataDirect Connect.

- a. Führen Sie die Prozedur `/opt/odbc/odbc.sh` aus. Mit dieser Prozedur werden eine Reihe von Umgebungsvariablen eingerichtet, die für das Betriebssystem spezifisch sind.
- b. Testen Sie die Verbindung zur Microsoft SQL Server-Datenquelle mit Hilfe des ODBC-Tools **demoodbc** von DataDirect Connect. Das Tool **demoodbc** befindet sich im Unterverzeichnis `/demo` der Connect ODBC-Bibliotheken.

Die nächste Task in dieser Tasksequenz ist das Festlegen der Umgebungsvariablen für Microsoft SQL Server.

#### Zugehörige Tasks:

- „Registrieren des Microsoft SQL Server-Wrappers“ auf Seite 336

## Festlegen der Umgebungsvariablen für Microsoft SQL Server

Das Festlegen der Microsoft SQL Server-Umgebungsvariablen ist ein Teil der umfangreicheren Task des Hinzufügens von Microsoft SQL Server zu einem Server mit zusammengeschlossenen Datenbanken.

Bei der Installation von DB2 Information Integrator versucht der Installationsprozess, die Microsoft SQL Server-Umgebungsvariablen in der Datei `db2dj.ini` festzulegen.

Die Umgebungsvariablen werden nicht in der Datei `db2dj.ini` festgelegt, wenn Sie den Microsoft SQL Server-ODBC-Treiber nach der Installation von DB2 Information Integrator installieren.

Gültige Umgebungsvariablen für Microsoft SQL Server sind:

- `DJX_ODBC_LIBRARY_PATH`
- `ODBCINI`
- `LD_LIBRARY_PATH` (nur Solaris)
- `SHLIB_PATH` (nur HP-UX)
- `DB2LIBPATH`
- `DB2ENVLIST`

Wird Ihr Server mit zusammengeschlossenen Datenbanken unter HP-UX ausgeführt und es besteht eine Exemplarkonfiguration mit mehreren Partitionen, müssen Sie die Werte `SHLIB_PATH` im Benutzerprofil exportieren.

#### Voraussetzungen:

Ein Server mit zusammengeschlossenen Datenbanken, der für den Zugriff auf Ihre Datenquellen korrekt konfiguriert ist. Dies beinhaltet die Installation und Konfiguration von erforderlicher Software, z. B. der Client-Software der Datenquelle.

#### Einschränkungen:

Siehe hierzu den Abschnitt: Einschränkungen für die Datei `db2dj.ini`

#### Vorgehensweise:

Gehen Sie wie folgt vor, um die Umgebungsvariablen für den Microsoft SQL Server-ODBC-Treiber automatisch festzulegen:

1. Installieren Sie den Microsoft SQL Server-ODBC-Treiber auf dem DB2-Server mit zusammengeschlossenen Datenbanken, falls noch nicht geschehen.

2. Legen Sie die erforderlichen Umgebungsvariablen fest. Sie können die Umgebungsvariablen automatisch festlegen, indem Sie die Installation von DB2 Information Integrator erneut ausführen. Klicken Sie in der Klickstartleiste **Produkte installieren an**, und folgen Sie den Anweisungen des Assistenten.

Gehen Sie wie folgt vor, um die Umgebungsvariablen für den Microsoft SQL Server-ODBC-Treiber manuell festzulegen:

1. Bearbeiten Sie die Datei `db2dj.ini`.
  - Auf unter Windows ausgeführten Servern mit zusammengeschlossenen Datenbanken befindet sich diese Datei im Verzeichnis `sql1lib/cfg`.
  - Auf unter UNIX ausgeführten Servern mit zusammengeschlossenen Datenbanken befindet sich diese Datei im Verzeichnis `sql1lib/cfg`.

Die Datei `db2dj.ini` enthält Konfigurationsinformationen über den Microsoft SQL Server-ODBC-Treiber, der auf Ihrem Server mit zusammengeschlossenen Datenbanken installiert ist. Ist die Datei nicht vorhanden, können Sie mit Hilfe eines beliebigen Texteditors eine neue Datei mit dem Namen `db2dj.ini` erstellen. In der Datei `db2dj.ini` müssen Sie den vollständig qualifizierten Pfad für die Variablen angeben, andernfalls treten Fehler auf.

2. Legen Sie die folgenden Umgebungsvariablen wie erforderlich fest:

#### **DJX\_ODBC\_LIBRARY\_PATH**

Setzen Sie den Verzeichnispfad auf die ODBC-Bibliotheksdateien. Beispiel:

```
DJX_ODBC_LIBRARY_PATH=ODBC-treiberverzeichnis/lib
```

ODBC-treiberverzeichnis ist der Verzeichnispfad, in dem der ODBC-Treiber installiert ist.

#### **ODBCINI**

Setzen Sie die Umgebungsvariable ODBCINI auf den Verzeichnispfad, in dem sich Ihre ODBC-Konfigurationsdatei (`odbc.ini`) befindet. Legen Sie die Umgebungsvariable ODBCINI nicht als Systemvariable fest. Beispiel:

```
ODBCINI=/home/db2inst1/.odbc.ini
```

#### **LD\_LIBRARY\_PATH**

Setzen Sie unter Solaris den Verzeichnispfad auf die ODBC-Bibliotheksdateien. Beispiel:

```
LD_LIBRARY_PATH=ODBC-treiberverzeichnis/lib
```

#### **SHLIB\_PATH**

Setzen Sie unter HP-UX den Verzeichnispfad auf die ODBC-Bibliotheksdateien. Beispiel:

```
SHLIB_PATH=ODBC-treiberverzeichnis/lib
```

3. Um auf Microsoft SQL Server zugreifen zu können, müssen Sie den Verzeichnispfad auf die ODBC-Bibliotheksdateien im Unterverzeichnis `lib` setzen. Beispiel:

```
db2set DB2LIBPATH=ODBC-treiberverzeichnis/lib
```

4. Um den Connect-ODBC-Treiber für den Zugriff auf Microsoft SQL Server-Datenquellen zu verwenden, setzen Sie `DB2ENVLIST` auf den Wert `LIBPATH`. Beispiel:

```
db2set DB2ENVLIST=LIBPATH
```

`LIBPATH` ist der Verzeichnispfad, in dem der ODBC-Treiber installiert ist.

5. Um sicherzustellen, dass die Umgebungsvariablen für den Server mit zusammengeschlossenen Datenbanken gesetzt sind, beenden Sie das DB2-Exemplar, und starten Sie es erneut. Geben Sie die folgenden Befehle aus, um das DB2-Exemplar erneut zu starten:

```
db2stop  
db2start
```

#### Für Exemplarkonfigurationen mit mehreren Partitionen unter HP-UX:

1. Erstellen Sie ein Benutzerprofil, falls noch keines vorhanden ist. Das Benutzerprofil befindet sich im Verzeichnis `$HOME/sqllib/`.
2. Fügen Sie `SHLIB_PATH` zum Benutzerprofil hinzu. Der Wert für `SHLIB_PATH` ist der Verzeichnispfad, in dem der ODBC-Treiber installiert ist. Geben Sie beispielsweise folgenden Befehl aus:

```
export SHLIB_PATH=$SHLIB_PATH:/home/DataDirectODBC/lib
```

Die nächste Task in dieser Tasksequenz ist das Registrieren des Microsoft SQL Server-Wrappers.

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für Microsoft SQL Server-Tabellen und Sichten“ auf Seite 344

#### Zugehörige Referenzen:

- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68
- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68

## Registrieren des Microsoft SQL Server-Wrappers

Die Registrierung des Microsoft SQL Server-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von Microsoft SQL Server-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken. Sie müssen einen Wrapper für den Zugriff auf Microsoft SQL Server-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

#### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung `CREATE WRAPPER` aus.

Beispiel:

```
CREATE WRAPPER MSSQLODBC3
```

**Empfehlung:** Verwenden Sie den standardmäßigen Wrappernamen `MSSQLODBC3`. Registrieren Sie einen Wrapper mit dem Standardnamen, verwendet der Server mit zusammengeschlossenen Datenbanken automatisch den standardmäßigen Bibliotheksnamen, der diesem Wrapper zugeordnet ist. Sollte zwischen dem Namen des Wrappers und einem bereits vorhandenen Wrappernamen in der zusammengeschlossenen Datenbank ein Namenskonflikt entstehen, können Sie den Standardnamen für den Wrapper durch einen beliebigen Namen ersetzen. Verwenden Sie einen anderen Namen als den Standardnamen, müssen Sie den Parameter `LIBRARY` in die Anweisung `CREATE WRAPPER` mit einschließen.

Um beispielsweise einen Wrapper mit dem Namen `sqlserver_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER sqlserver_wrapper LIBRARY 'libdb2mssql3.a'
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung `CREATE WRAPPER` finden Sie in der Liste der Bibliotheksdateien von Microsoft SQL-Server-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den Microsoft SQL Server-Wrapper.

**Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine Microsoft SQL Server-Datenquelle“ auf Seite 338

**Zugehörige Referenzen:**

- „Bibliotheksdateien des Microsoft SQL Server-Wrappers“ auf Seite 337
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien des Microsoft SQL Server-Wrappers

Die folgende Tabelle listet die Verzeichnispfade und Bibliotheksdateinamen für den Microsoft SQL Server-Wrapper auf.

Wenn Sie DB2 Information Integrator installieren, werden 3 Bibliotheksdateien zu dem Verzeichnispfad hinzugefügt, der in der Tabelle aufgeführt ist. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden folgende Wrapperbibliotheksdateien zum Verzeichnispfad hinzugefügt: `libdb2mssql3.a`, `libdb2mssql3F.a` und `libdb2mssql3U.a`.

Geben Sie beim Registrieren eines Wrappers ausschließlich den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 77. Positionen und Dateinamen von Microsoft SQL Server-Clientbibliotheken*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	<code>/usr/opt/db2_08_01/lib/</code>	<code>libdb2mssql3.a</code>
HP-UX	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2mssql3.sl</code>
Linux	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2mssql3.so</code>
Solaris	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2mssql3.so</code>
Windows	<code>%DB2PATH%\bin</code>	<code>db2mssql3.dll</code>

`%DB2PATH%` ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows ist `C:\Program Files\IBM\SQLLIB`.

**Zugehörige Tasks:**

- „Registrieren des Microsoft SQL Server-Wrappers“ auf Seite 336

## Registrieren der Serverdefinitionen für eine Microsoft SQL Server-Datenquelle

Beim Hinzufügen von Microsoft SQL Server-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen die Serverdefinitionen für eine Microsoft SQL Server-Datenquelle registriert werden.

In der zusammengeschlossenen Datenbank müssen Sie jeden fernen Server mit Microsoft SQL Server definieren, auf den Sie zugreifen möchten. Zunächst müssen Sie den Knotennamen des fernen Servers mit Microsoft SQL Server suchen und diesen Knotennamen anschließend verwenden, wenn Sie die Serverdefinition mit Hilfe der Anweisung CREATE SERVER registrieren.

### Vorgehensweise:

Um eine Serverdefinition für eine Microsoft SQL Server-Datenquelle zu registrieren, gehen Sie wie folgt vor:

1. Suchen Sie den Knotennamen.
  - Wenn Ihr Server mit zusammengeschlossenen Datenbanken Windows verwendet, so ist NODE der System-DSN-Name, den Sie für den fernen Server von Microsoft SQL Server angegeben haben, auf den Sie zugreifen.
  - Verwendet Ihr Server mit zusammengeschlossenen Datenbanken UNIX, so wird NODE in der Datei .odbc.ini angegeben.

Nachfolgend ist ein Beispiel für die Datei .odbc.ini unter AIX aufgeführt.

### Beispiel für Datei '.odbc.ini' unter AIX:

```
rawilson=MS SQL Server 7.0
medusa=MS SQL Server 7.0
[rawilson]
Driver=/opt/odbc/lib/ivmsss16.so
Description=MS SQL Server Driver for AIX
Address=9.112.30.39,1433
[medusa]
Driver=/opt/odbc/lib/ivmsss16.so
Description=MS SQL Server Driver for AIX
Address=9.112.98.123,1433
```

Am Anfang der Datei .odbc.ini befindet sich ein Abschnitt, der mit [ODBC Data Sources] gekennzeichnet ist und die Knoten auflistet. Jeder der Knoten verfügt über einen Abschnitt [node\_name], in dem er beschrieben wird.

Obwohl der Knotenname in der Anweisung CREATE SERVER als Option angegeben wird, ist er für Microsoft SQL Server-Datenquellen erforderlich.

2. Setzen Sie die Anweisung CREATE SERVER ab. Beispiel:

```
CREATE SERVER servername TYPE MSSQLSERVER VERSION 7.0 WRAPPER mssqlodb3
OPTIONS (NODE 'sqlnode', DBNAME 'mssdb');
```

Verwenden Sie nach dem Erstellen der Serverdefinition die Anweisung ALTER SERVER, um Serveroptionen hinzuzufügen oder zu löschen.

Die nächste Task in dieser Tasksequenz ist das Erstellen einer Benutzerzuordnung für eine Microsoft SQL Server-Datenquelle.

### Zugehörige Tasks:

- „Erstellen einer Benutzerzuordnung für eine Microsoft SQL Server-Datenquelle“ auf Seite 341



### Zugehörige Referenzen:

- „ALTER SERVER statement“ in *SQL Reference, Volume 2*
- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641
- „Anweisung CREATE SERVER - Beispiele für den Microsoft SQL Server-Wrapper“ auf Seite 339

## Anweisung CREATE SERVER - Beispiele für den Microsoft SQL Server-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE SERVER verwendet wird, um Server für den Microsoft SQL Server-Wrapper zu registrieren. Dieser Abschnitt enthält ein vollständiges Beispiel, das zeigt, wie ein Server mit den erforderlichen Parametern registriert wird, sowie ein Beispiel mit zusätzlichen Serveroptionen.

### Vollständiges Beispiel:

Das folgende Beispiel zeigt, wie eine Serverdefinition für einen Microsoft SQL Server-Wrapper mit Hilfe der Anweisung CREATE SERVER registriert wird:

```
CREATE SERVER sqlserver TYPE MSSQLSERVER VERSION 7.0 WRAPPER mssqlodbc3  
OPTIONS (NODE 'sqlnode', DBNAME 'africa');
```

*sqlserver*

Ein Name, der dem fernen Server mit Microsoft SQL Server zugeordnet wird. Dieser Name muss eindeutig sein. Doppelte Servernamen sind nicht zulässig.

**TYPE** *MSSQLSERVER*

Der Typ der Datenquelle, für den der Zugriff konfiguriert wird. Der Parameter TYPE für den Microsoft SQL Server-Wrapper muss *MSSQLSERVER* lauten.

**VERSION** *7.0*

Die Version der Microsoft SQL Server-Datenbankserversoftware, auf die zugegriffen werden soll. Die Versionen 6.5, 7.0 und 2000 werden unterstützt.

**WRAPPER** *mssqlodbc3*

Der Wrapper-Name, der in der Anweisung CREATE WRAPPER angegeben wurde.

**NODE** '*sqlnode*'

Unter Windows der System-DSN-Name für den fernen Server von Microsoft SQL Server, auf den Sie zugreifen. Unter UNIX der Knoten, der in der Datei *.odbc.ini* angegeben wird. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

Obwohl der Name des Knotens in der Anweisung CREATE SERVER als Option angegeben wird, ist er für Microsoft SQL Server-Datenquellen erforderlich.



**DBNAME 'africa'**

Der Name der Datenbank, auf die zugegriffen werden soll. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

Obwohl der Name der Datenbank in der Anweisung CREATE SERVER als Option angegeben wird, ist er für Microsoft SQL Server-Datenquellen erforderlich.

**Beispiele mit Serveroptionen:**

Wenn Sie den Server registrieren, können Sie in der Anweisung CREATE SERVER zusätzliche Serveroptionen angeben. Diese Serveroptionen umfassen allgemeine Serveroptionen sowie Serveroptionen, die für Microsoft SQL Server spezifisch sind.

Das folgende Beispiel zeigt, wie die Serveroption COLLATING\_SEQUENCE verwendet wird:

```
CREATE SERVER sqlserver TYPE MSSQLSERVER
  VERSION 7.0
  WRAPPER mssqlodbc3
  OPTIONS (NODE 'sqlnode', DBNAME 'africa', COLLATING_SEQUENCE 'I');
```

Die Serveroption COLLATING\_SEQUENCE gibt an, ob die Datenquelle dieselbe Sortierfolge verwendet wie der Server mit zusammengeschlossenen Datenbanken. Auf einem Microsoft SQL Server-Datenbankserver mit dem Betriebssystem Windows NT oder Windows 2000 muss bei der standardmäßigen Sortierfolge die Groß-/Kleinschreibung nicht beachtet werden (beispielsweise gelten 'STEWART' und 'StewART' als gleich). Um korrekte Ergebnisse vom Server mit zusammengeschlossenen Datenbanken sicherzustellen, müssen Sie die Serveroption COLLATING\_SEQUENCE auf 'I' setzen. Diese Einstellung gibt an, dass die Microsoft SQL Server-Datenquelle nicht von der Groß-/Kleinschreibung abhängt.

Der Server mit zusammengeschlossenen Datenbanken führt keine Pushdown-Verarbeitung für Abfragen durch, wenn sich die von der Datenquelle zurückgegebenen Ergebnisse von den Ergebnissen unterscheiden, die bei der Verarbeitung der Abfrage auf dem Server mit zusammengeschlossenen Datenbanken zurückgegeben werden. Wenn Sie die Serveroption COLLATING\_SEQUENCE auf 'I' setzen, führt der Server mit zusammengeschlossenen Datenbanken keine Pushdown-Verarbeitung durch für Abfragen mit Zeichenfolgedaten oder Ausdrücken sowie für Abfragen, die folgende Klauseln, Vergleichselemente oder Funktionen enthalten:

- GROUP BY-Klauseln
- DISTINCT-Klauseln
- BASIC-Vergleichselemente wie beispielsweise 'gleich' (=)
- Spaltenfunktionen wie beispielsweise MIN oder MAX

**Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine Microsoft SQL Server-Datenquelle“ auf Seite 338

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

## Erstellen einer Benutzerzuordnung für eine Microsoft SQL Server-Datenquelle

Beim Hinzufügen von Microsoft SQL Server-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken muss eine Benutzerzuordnung für eine Microsoft SQL Server-Datenquelle erstellt werden.

Wird versucht, auf eine Microsoft SQL Server-Datenquelle zuzugreifen, stellt der Server mit zusammengeschlossenen Datenbanken eine Verbindung zur Datenquelle her. Hierbei verwendet er eine Benutzer-ID und ein Kennwort, die für die betreffende Datenquelle gültig sind. Sie müssen eine Zuordnung (eine Benutzerzuordnung) zwischen jeder Benutzer-ID sowie jedem Kennwort des Servers mit zusammengeschlossenen Datenbanken und der entsprechenden Benutzer-ID und dem entsprechenden Kennwort der Datenquelle definieren. Erstellen Sie eine Benutzerzuordnung für jede Benutzer-ID, über die auf das System zusammengeschlossener Datenbanken zugegriffen wird, um verteilte Anforderungen an die Informix-Datenquelle zu senden.

### Vorgehensweise:

Um der Benutzer-ID und dem Kennwort des fernen Servers mit Microsoft SQL Server eine lokale Benutzer-ID zuzuordnen, setzen Sie die Anweisung CREATE USER MAPPING ab.

Beispiel:

```
CREATE USER MAPPING FOR benutzer-id SERVER sqlserver
    OPTIONS (REMOTE_AUTHID 'ferne_id', REMOTE_PASSWORD 'fernes_kennwort)
```

Die nächste Task in dieser Tasksequenz ist das Testen der Verbindung zum fernen Server mit Microsoft SQL Server.

### Zugehörige Tasks:

- „Testen der Verbindung zum fernen Server mit Microsoft SQL Server“ auf Seite 342

### Zugehörige Referenzen:

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE USER MAPPING - Beispiele für den Microsoft SQL Server-Wrapper“ auf Seite 341

## Anweisung CREATE USER MAPPING - Beispiele für den Microsoft SQL Server-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE USER MAPPING verwendet wird, um die Benutzer-ID eines Servers mit zusammengeschlossenen Datenbanken einer Benutzer-ID und einem Kennwort eines fernen Servers mit Microsoft SQL Server zuzuordnen. Dieser Abschnitt enthält ein vollständiges Beispiel mit den erforderlichen Parametern sowie ein Beispiel, das zeigt, wie das DB2-Sonderregister USER mit der Anweisung CREATE USER MAPPING verwendet wird.

### Vollständiges Beispiel:

Das folgende Beispiel zeigt, wie die Benutzer-ID eines Servers mit zusammengesetzten Datenbanken der Benutzer-ID und dem Kennwort eines fernen Servers mit Microsoft SQL Server zugeordnet wird:

```
CREATE USER MAPPING FOR elizabeth SERVER sqlserver
    OPTIONS (REMOTE_AUTHID 'liz', REMOTE_PASSWORD 'abc123')
```

*elizabeth*

Gibt die lokale Benutzer-ID an, die Sie einer Benutzer-ID zuordnen, die auf dem fernen Server mit Microsoft SQL Server definiert wurde.

**SERVER** *sqlserver*

Gibt den Namen des fernen Servers mit Microsoft SQL Server an, den Sie in der Anweisung CREATE SERVER definiert haben.

**REMOTE\_AUTHID** '*liz*'

Gibt die Benutzer-ID auf dem fernen Server mit Microsoft SQL Server an, zu der Sie *elizabeth* zuordnen. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption FOLD\_ID in der Anweisung CREATE SERVER nicht auf 'U' oder 'L' setzen.

**REMOTE\_PASSWORD** '*abc123*'

Gibt das Kennwort an, das '*liz*' zugeordnet ist. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption FOLD\_PW in der Anweisung CREATE SERVER nicht auf 'U' oder 'L' setzen.

### Beispiel für das Sonderregister:

Das folgende Beispiel zeigt eine Anweisung CREATE USER MAPPING, die das Sonderregister USER enthält:

```
CREATE USER MAPPING FOR USER SERVER sqlserver
    OPTIONS (REMOTE_AUTHID 'liz', REMOTE_PASSWORD 'abc123')
```

Sie können das DB2-Sonderregister USER verwenden, um die Berechtigungs-ID der Person, die die Anweisung CREATE USER MAPPING absetzt, zur Berechtigungs-ID der Datenquelle zuzuordnen, die in der Benutzeroption REMOTE\_AUTHID angegeben wurde.

### Zugehörige Tasks:

- „Erstellen einer Benutzerzuordnung für eine Microsoft SQL Server-Datenquelle“ auf Seite 341

### Zugehörige Referenzen:

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

## Testen der Verbindung zum fernen Server mit Microsoft SQL Server

Beim Hinzufügen von Microsoft SQL Server-Datenquellen zu Servern mit zusammengesetzten Datenbanken muss die Verbindung zum fernen Server mit Microsoft SQL Server getestet werden. Sie können die Verbindung zum fernen Server mit Microsoft SQL Server testen, indem Sie die Serverdefinition und Benutzerzuordnungen verwenden, die Sie definiert haben.

### Vorgehensweise:

Um die Verbindung zu testen, gehen Sie wie folgt vor:

1. Öffnen Sie eine Durchgriffssitzung, um eine SQL-Anweisung SELECT für die Microsoft SQL Server-Systemtabellen abzusetzen.

Beispiel:

```
SET PASSTHRU name_des_fernen_servers
SELECT count(*) FROM dbo.sysobjects
SET PASSTHRU RESET
```

Wenn die SQL-Anweisung SELECT eine Anzahl zurückgibt, sind Ihre Serverdefinition und Ihre Benutzerzuordnung ordnungsgemäß eingerichtet.

2. Wenn die SQL-Anweisung SELECT einen Fehler zurückgibt, müssen Sie möglicherweise folgende Aktionen ausführen:
  - Prüfen des fernen Servers mit Microsoft SQL Server, um sicherzustellen, dass er gestartet wurde.
  - Prüfen des fernen Servers mit Microsoft SQL Server, um sicherzustellen, dass er für ankommende Verbindungsanforderungen konfiguriert wurde.
  - Prüfen Ihrer Benutzerzuordnung, um sicherzustellen, dass die Einstellungen für die Optionen REMOTE\_AUTHID und REMOTE\_PASSWORD für Verbindungen zum fernen Server mit Microsoft SQL Server gültig sind. Ändern Sie gegebenenfalls die Benutzerzuordnung oder erstellen Sie gegebenenfalls eine andere Benutzerzuordnung.
  - Prüfen der ODBC-Treiber auf dem DB2-Server mit zusammengeschlossenen Datenbanken, um sicherzustellen, dass sie für das Herstellen einer Verbindung zum fernen Server mit Microsoft SQL Server ordnungsgemäß installiert und konfiguriert wurden.
  - Prüfen der Einstellungen Ihrer DB2-Variablen für zusammengeschlossene Datenbanken, um sicherzustellen, dass sie für das Arbeiten mit dem fernen Server mit Microsoft SQL Server korrekt definiert sind. Zu diesen Variablen gehören die Systemumgebungsvariablen, die db2dj.ini-Variablen und die Variable der DB2-Profilregistrierdatenbank (db2set).
  - Prüfen Ihrer Serverdefinition. Gegebenenfalls müssen Sie diese Definition löschen und erneut erstellen.

Die nächst Task in dieser Tasksequenz ist das Registrieren von Kurznamen für Microsoft SQL Server-Tabellen und -Sichten.

### Zugehörige Tasks:

- „Hinzufügen von Microsoft SQL Server-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 331
- „Registrieren von Kurznamen für Microsoft SQL Server-Tabellen und Sichten“ auf Seite 344
- „Festlegen der Umgebungsvariablen für Microsoft SQL Server“ auf Seite 334

### Zugehörige Referenzen:

- „ALTER USER MAPPING statement“ in *SQL Reference, Volume 2*

## Registrieren von Kurznamen für Microsoft SQL Server-Tabellen und Sichten

Beim Hinzufügen von Microsoft SQL Server-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen Kurznamen für Microsoft SQL Server-Tabellen und -Sichten registriert werden.

Registrieren Sie für jeden fernen Server mit Microsoft SQL Server, den Sie definieren, jeweils einen Kurznamen für alle Tabellen oder Sichten, auf die Sie zugreifen wollen. Verwenden Sie diese Kurznamen anstelle der Namen der Datenquellenobjekte, wenn Sie die fernen Server mit Microsoft SQL Server abfragen.

Die zusammengeschlossene Datenbank verwendet Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Diese Statistiken werden gesammelt, wenn Sie unter Verwendung der Anweisung `CREATE NICKNAME` einen Kurznamen für ein Datenquellenobjekt registrieren. Die zusammengeschlossene Datenbank stellt das Vorhandensein des Objekts in der Datenquelle sicher und versucht dann, bestehende statistische Daten zur Datenquelle zu sammeln. Informationen, die für das Optimierungsprogramm nützlich sind, werden aus den Katalogen der Datenquelle gelesen und in den globalen Katalog auf dem Server mit zusammengeschlossenen Datenbanken aufgenommen. Da einige oder alle der Kataloginformationen von Datenquellen vom Optimierungsprogramm verwendet werden können, sollten Sie die Statistiken (unter Verwendung des Datenquellenbefehls, der dem DB2-Befehl **RUNSTATS** entspricht) in der Datenquelle aktualisieren, bevor Sie einen Kurznamen registrieren.

### Vorgehensweise:

Um einen Kurznamen zu registrieren, setzen Sie die Anweisung `CREATE NICKNAME` ab.

Beispiel:

```
CREATE NICKNAME mss-name FOR sqlserver."remote_schema"."remote.table"
```

Kurznamen können bis zu 128 Zeichen lang sein.

Wiederholen Sie diesen Schritt für jede Microsoft SQL Server-Tabelle oder -Sicht, für die Sie Kurznamen erstellen möchten.

Wenn Sie den Kurznamen erstellen, verwendet DB2 die Verbindung dazu, die Tabellen des Datenquellenkatalogs abzufragen (Microsoft SQL Server nennt diese Tabellen Systemtabellen). Diese Abfrage testet Ihre Verbindung zur Datenquelle unter Verwendung des Kurznamens. Kann keine Verbindung hergestellt werden, erhalten Sie eine Fehlermeldung.

### Zugehörige Referenzen:

- „RUNSTATS Command“ in *Command Reference*
- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den Microsoft SQL Server-Wrapper“ auf Seite 345

## Anweisung CREATE NICKNAME - Beispiele für den Microsoft SQL Server-Wrapper

Dieser Abschnitt enthält ein Beispiel, das zeigt, wie die Anweisung CREATE NICKNAME verwendet wird, um einen Kurznamen für Microsoft SQL Server-Tabellen oder -Sichten zu registrieren, auf die Sie zugreifen möchten.

Dieses Beispiel zeigt, wie ein fernes Objekt für den fernen Server mit Microsoft SQL Server angegeben wird, unter dem der Kurzname zugeordnet ist:

```
CREATE NICKNAME cust_africa FOR sqlserver.customers.egypt
```

*cust\_africa*

Ein eindeutiger Kurzname für die Microsoft SQL Server-Tabelle oder -Sicht.

**Anmerkung:** Der Kurzname besteht aus zwei Teilen - dem Schema und dem Kurznamen. Wenn Sie beim Registrieren des Kurznamens das Schema auslassen, wird die Berechtigungs-ID des Benutzers, der den Kurznamen erstellt, zum Schema des Kurznamens.

*sqlserver.customers.egypt*

Eine dreiteilige Kennung für das ferne Objekt.

- *sqlserver* ist der Name, den Sie dem Microsoft SQL Server-Datenbankserver in der Anweisung CREATE SERVER zugeordnet haben.
- *customers* ist der Name des fernen Schemas, zu dem die Tabelle oder Sicht gehört.
- *egypt* ist der Name der fernen Tabelle oder Sicht, auf die Sie zugreifen möchten.

Der Server mit zusammengeschlossenen Datenbanken setzt die Namen der Microsoft SQL Server-Schemata und -Tabellen in Großbuchstaben um, sofern Sie die Namen nicht in Anführungszeichen setzen.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für Microsoft SQL Server-Tabellen und Sichten“ auf Seite 344

### Zugehörige Referenzen:

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

---

## Optimieren der Konfiguration für Microsoft SQL Server-Datenquellen und Beheben von Konfigurationsfehlern

Nachdem Sie die Konfiguration für Microsoft SQL Server-Datenquellen eingerichtet haben, müssen Sie diese Konfiguration möglicherweise ändern, um die Leistung zu verbessern.

### Abrufen von ODBC-Traces

Wenn beim Zugriff auf die Datenquelle Probleme auftreten, können Sie ODBC-Traceinformationen abrufen, um diese Probleme zu analysieren und zu lösen. Das Aktivieren eines Trace wirkt sich jedoch auf die Systemleistung aus. Sie sollten die Tracefunktion wieder ausschalten, nachdem Sie die Probleme gelöst haben.

Verwenden Sie bei Servern mit zusammengeschlossenen Datenbanken unter Windows das über das Fenster 'ODBC-Datenquellen-Administrator' bereitgestellte Trace-Tool, um sicherzustellen, dass die Ausführung von ODBC-Traces ordnungsgemäß funktioniert.

Bei Servern mit zusammengeschlossenen Datenbanken unter UNIX können Sie die Tracefunktion durch Ändern der Datei `odbc.ini` aktivieren. Wenn Sie z. B. den Treiber DataDirect ODBC 3.x verwenden, suchen Sie das Beispiel für die Datei `odbc.ini` im Clientverzeichnis. Die Datei `odbc.ini` enthält ein Beispiel für die Anforderungen der Tracedateien:

```
[ODBC]
Trace=0
TraceFile=/home/user1/trace_dir/filename.xxx
TraceDll==ODBC-treiberverzeichnis/odbctrac.so
InstallDir=/opt/odbc
```

Um die Tracefunktion zu aktivieren, setzen Sie die erste Zeile auf `Trace=1`. Um die Tracefunktion zu inaktivieren, setzen Sie die erste Zeile auf `Trace=0`. Die Datei `TraceFile` sollte auf einen Pfad und Dateinamen mit Schreibzugriff für das Exemplar der zusammengeschlossenen Datenbank zeigen.

#### **Zugehörige Tasks:**

- „Hinzufügen von Microsoft SQL Server-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 331

#### **Zugehörige Referenzen:**

- „db2set - DB2 Profile Registry Command“ in *Command Reference*



---

## Kapitel 17. Konfigurieren des Zugriffs auf ODBC-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in ODBC-Datenquellen gespeichert sind. Der Zugriff auf ODBC-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält eine Liste der Tasks, die Sie ausführen müssen, wenn Sie den Server mit zusammengeschlossenen Datenbanken konfigurieren, zeigt Beispiele zu den benötigten SQL-Anweisungen und bietet Informationen zur Optimierung und Fehlerbehebung beim Konfigurieren des Servers mit zusammengeschlossenen Datenbanken.

---

### Hinzufügen von ODBC zu einem System zusammengeschlossener Datenbanken

#### Hinzufügen von ODBC-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf ODBC-Datenquellen zu konfigurieren, müssen Sie dem Server Informationen zu den Datenquellen und -objekten zur Verfügung stellen, auf die Sie zugreifen möchten.

Der Zugriff des Servers mit zusammengeschlossenen Datenbanken auf ODBC-Datenquellen kann über die DB2-Steuerzentrale oder die DB2-Befehlszeile konfiguriert werden. Die DB2-Steuerzentrale umfasst einen Assistenten, der Sie durch die erforderlichen Schritte zum Konfigurieren des Servers mit zusammengeschlossenen Datenbanken führt.

Datenquellen, auf die über die ODBC-API zugegriffen wird, werden in diesem Text als ODBC-Datenquellen bezeichnet.

Entsprechend Ihren Anforderungen können Sie auf Excel-Daten mit dem ODBC-Wrapper zugreifen, anstatt den Excel-Wrapper zu verwenden. Die spezifischen Schritte zum Konfigurieren des ODBC-Wrappers für den Zugriff auf Excel-Daten sind in einem eigenen Abschnitt dokumentiert.

#### Voraussetzungen:

- DB2 Information Integrator muss auf einem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss auf dem Server mit zusammengeschlossenen Datenbanken vorhanden sein.
- Der ODBC-Treiber muss auf dem Server mit zusammengeschlossenen Datenbanken installiert und konfiguriert sein. Der ODBC-Wrapper unterstützt ODBC 3.x.
- Ordnungsgemäße Einrichtung der Systemumgebungsvariablen, der db2dj.ini-Variablen und der Variablen der DB2-Profilregistrierdatenbank (db2set). Die Dokumentation des Lieferanten enthält die erforderlichen Variablen für Ihren ODBC-Client. Unter Umständen ist die Variable LIBPATH erforderlich.



### Einschränkungen:

- Mit dem ODBC-Wrapper kann nicht auf Datenquellen der DB2-Produktfamilie zugegriffen werden. Für den Zugriff auf Datenquellen der DB2-Produktfamilie muss der DRDA-Wrapper verwendet werden.
- Der ODBC-Wrapper bietet keine Unterstützung für die folgenden Funktionen und Anweisungen:
  - LOCK TABLE-Anweisungen für Kurznamen
  - Funktionen, die in ODBC 3.x nicht mehr unterstützt werden
  - X/Open- oder SQL/CLI-Treiber
  - Kurznamen für gespeicherte Prozeduren
  - Erzwingen von Ganzheitlichkeit auf Anweisungsebene mit Anweisungen für ferne Sicherungspunkte
  - 64-Bit-Clients
  - WITH HOLD-Cursor
- Bei Datenquellen, die keine positionierten Aktualisierungs- und Löschoptionen unterstützen, schlagen positionierte UPDATE- und DELETE-Anweisungen sowie bestimmte UPDATE- und DELETE-Anweisungen mit Suche für einen Kurznamen fehl, wenn für den Kurznamen oder seine entsprechende ferne Tabelle kein eindeutiger Index für Spalten vorhanden ist, die keine Nullzeichen enthalten dürfen. Wenn diese Anweisungen fehlschlagen, wird der Fehler SQL30090 mit Ursachencode 21 zurückgegeben.
- Der ODBC-Wrapper unterstützt keine INSERT-, UPDATE- und DELETE-Anweisungen für Datenquellen, bei denen die Anzahl der aktiven Anweisungen für die einzelnen Verbindungen beschränkt ist. Die Dokumentation zur betreffenden Datenquelle enthält Informationen dazu, ob die Anzahl der aktiven Anweisungen für die einzelnen Verbindungen bei dieser Datenquelle beschränkt ist. Diese Einschränkung gilt für IBM Red Brick Warehouse.

### Vorgehensweise:

Um ODBC-Datenquellen einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen, gehen Sie wie folgt vor:

1. Bereiten Sie den Server mit zusammengeschlossenen Datenbanken und die zusammengeschlossene Datenbank vor.
2. Registrieren Sie den Wrapper.
3. Registrieren Sie die Serverdefinition.
4. Erstellen Sie die Benutzerzuordnungen.
5. Testen Sie die Verbindung zur ODBC-Datenquelle.
6. Registrieren Sie Kurznamen für ODBC-Datenquellentabellen und -sichten.

### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63
- „Methoden für den Zugriff auf Excel-Daten“ auf Seite 30

### Zugehörige Tasks:

- „Zugreifen auf Excel-Daten mit dem ODBC-Wrapper“ auf Seite 359
- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58

- „Vorbereiten des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen über ODBC“ auf Seite 349
- „Registrieren des ODBC-Wrappers“ auf Seite 350
- „Registrieren der Serverdefinitionen für eine ODBC-Datenquelle“ auf Seite 352
- „Erstellen einer Benutzerzuordnung für eine ODBC-Datenquelle“ auf Seite 354
- „Testen der Verbindung zum ODBC-Datenquellenserver“ auf Seite 356
- „Registrieren von Kurznamen für ODBC-Datenquellentabellen und -sichten“ auf Seite 357

**Zugehörige Referenzen:**

- „Auswählen des korrekten Wrappers“ auf Seite 29

## Vorbereiten des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen über ODBC

Beim Hinzufügen von ODBC-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken muss der Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen über ODBC vorbereitet werden.

Welche Schritte zur Vorbereitung des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen über ODBC erforderlich sind, hängt vom Betriebssystem auf Ihrem Server mit zusammengeschlossenen Datenbanken ab.

**Hinweis:** Der ODBC-Treiber und das verwendete Betriebssystem haben eindeutige Speicherpositionen für den Bibliothekspfad.

**Vorgehensweise:**

Um den Server mit zusammengeschlossenen Datenbanken vorzubereiten, gehen Sie wie folgt vor:

**Unter Windows::**

Prüfen Sie, ob der Treiber für ODBC 3.x auf dem Server mit zusammengeschlossenen Datenbanken installiert und konfiguriert ist. Der Knotenname für die ODBC-Datenquelle muss im System-DSN definiert werden. Die Dokumentation zum ODBC-Treiber enthält Informationen zur Vorgehensweise bei der Installation und Konfiguration.

Wurde der DSN mit Hilfe von Microsoft ODBC Data Source Administrator konfiguriert, kann die entsprechende Einstellung in der Systemsteuerung überprüft werden. Stellen Sie sicher, dass die ODBC-Datenquelle als System-DSN registriert ist, da DB2 den DSN ansonsten möglicherweise nicht finden kann.

**Unter UNIX:**

1. Die Anweisungen zum Konfigurieren des ODBC-Clients finden Sie in der Dokumentation des Lieferanten des ODBC-Clients.
2. Handelt es sich bei dem Client um DataDirect ODBC oder RedBrick, prüfen Sie, ob die entsprechende symbolische Verbindung erstellt wurde:
  - Unter HP-UX besteht die symbolische Verbindung von `/usr/exe/libodbcinst.sl` zu `$ODBC_LIBRARY_PATH/libodbcinst.sl`.
  - Unter Linux besteht die symbolische Verbindung von `/usr/lib/libodbcinst.so` zu `$DJX_ODBC_LIBRARY_PATH/libodbcinst.so`.

- Unter Solaris besteht die symbolische Verbindung von `$HOME/sql11b/locale` zu `$DJX_ODBC_LIBRARY_PATH/./locale`. `$HOME` ist das Ausgangsverzeichnis des DB2-Exemplareigners.

Die nächste Task in dieser Tasksequenz ist das Registrieren des ODBC-Wrappers.

#### Zugehörige Tasks:

- „Registrieren des ODBC-Wrappers“ auf Seite 350

## Registrieren des ODBC-Wrappers

Beim Hinzufügen von ODBC-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken muss der ODBC-Wrapper registriert werden.

Das Registrieren eines Wrappers ist für den Zugriff auf ODBC-Datenquellen erforderlich. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken verwendet, um mit Datenquellen zu kommunizieren und Daten aus Datenquellen abzurufen. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

#### Vorgehensweise:

Um einen Wrapper zu registrieren, setzen Sie die Anweisung `CREATE WRAPPER` ab und geben Sie den Standardnamen für den Wrapper an.

Beispiel:

```
CREATE WRAPPER ODBC
```

**Empfehlung:** Verwenden Sie den standardmäßigen Wrappernamen `ODBC`. Wenn Sie den Wrapper mit dem Standardnamen registrieren, übernimmt der Server mit zusammengeschlossenen Datenbanken automatisch den standardmäßigen Bibliotheksnamen, der diesem Wrapper zugeordnet ist.

Sollte zwischen dem Namen des Wrappers und einem bereits vorhandenen Wrappernamen in der zusammengeschlossenen Datenbank ein Namenskonflikt entstehen, können Sie den Standardnamen für den Wrapper durch einen beliebigen Namen ersetzen. In diesem Fall müssen Sie in der Anweisung `CREATE WRAPPER` den Parameter `LIBRARY` angeben.

Um beispielsweise einen Wrapper mit dem Namen `odbc_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken zu registrieren, der das Betriebssystem AIX verwendet, setzen Sie die folgende Anweisung ab:

```
CREATE WRAPPER odbc_wrapper  
LIBRARY 'libdb2rcodbc.a' OPTIONS (MODULE '/usr/lib/odbc.a')
```

`MODULE '/usr/lib/odbc.a'` ist der vollständige Pfad der Bibliothek, die den ODBC-Treibermanager (ODBC Driver Manager) enthält.

Der Name der von Ihnen angegebene Wrapperbibliotheksdatei hängt von dem Betriebssystem auf dem Server mit zusammengeschlossenen Datenbanken ab. Den korrekten Namen, der in der Anweisung `CREATE WRAPPER` angegeben werden muss, finden Sie in der Liste der ODBC-Wrapper-Bibliotheksdateien.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den ODBC-Wrapper.

#### Zugehörige Tasks:

- „Registrieren der Serverdefinitionen für eine ODBC-Datenquelle“ auf Seite 352

#### Zugehörige Referenzen:

- „ODBC-Wrapper-Bibliotheksdateien“ auf Seite 351
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE WRAPPER - Beispiele für den ODBC-Wrapper“ auf Seite 351

## ODBC-Wrapper-Bibliotheksdateien

Die folgende Tabelle enthält eine Liste der Verzeichnispfade und Bibliotheksdateinamen für den ODBC-Wrapper.

Wenn Sie DB2 Information Integrator installieren, werden dem in der Tabelle aufgeführten Verzeichnispfad 3 Bibliotheksdateien hinzugefügt. Wird der Server mit zusammengesetzten Datenbanken beispielsweise unter AIX ausgeführt, werden die Wrapperbibliotheksdateien `libdb2rcodbc.a`, `libdb2rcodbcF.a` und `libdb2rcodbcU.a` dem Verzeichnispfad hinzugefügt.

Wenn Sie einen Wrapper registrieren, geben Sie nur den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 78. Speicherpositionen und Dateinamen der ODBC-Wrapperbibliothek*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2rcodbc.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2rcodbc.sl
Linux	/opt/IBM/db2/V8.1/lib	libdb2rcodbc.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2rcodbc.so
Windows	%DB2PATH%\bin	db2rcodbc.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows lautet `C:\Program Files\IBM\SQLLIB`.

#### Zugehörige Tasks:

- „Registrieren des ODBC-Wrappers“ auf Seite 350

## Anweisung CREATE WRAPPER - Beispiele für den ODBC-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE WRAPPER verwendet wird, um Wrapper für ODBC-Datenquellen zu registrieren.

#### Beispiel für UNIX-Systeme:

Das folgende Beispiel zeigt, wie ein Wrapper mit Hilfe der Anweisung CREATE WRAPPER auf einem UNIX-Betriebssystem registriert wird:

```
CREATE WRAPPER odbc OPTIONS (MODULE '/usr/lib/odbc.so')
```

In diesem Beispiel ist *odbc* der Name, den Sie dem Wrapper zuordnen, der in der zusammengesetzten Datenbank registriert wird. `MODULE '/usr/lib/odbc.so'` ist der vollständige Pfad der Bibliothek, die ODBC Driver Manager enthält.

Auf UNIX-Betriebssystemen müssen Sie die Option `MODULE` angeben. Unter Windows nimmt die Option `MODULE` standardmäßig den Wert `'odbc32.dll'` an.

#### Beispiel für Windows:

Das folgende Beispiel zeigt, wie ein Wrapper mit Hilfe der Anweisung `CREATE WRAPPER` auf einem Windows-Betriebssystem registriert wird:

```
CREATE WRAPPER odbc LIBRARY 'db2rcodbc.dll'
```

In diesem Beispiel ist *odbc* der Name, den Sie dem Wrapper zuordnen, der in der zusammengesetzten Datenbank registriert wird. `LIBRARY 'db2rcodbc.dll'` ist der Bibliotheksname für den ODBC-Wrapper.

#### Zugehörige Tasks:

- „Registrieren des ODBC-Wrappers“ auf Seite 350

#### Zugehörige Referenzen:

- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Registrieren der Serverdefinitionen für eine ODBC-Datenquelle

Beim Hinzufügen von ODBC-Datenquellen zu Servern mit zusammengesetzten Datenbanken müssen die Serverdefinitionen für eine ODBC-Datenquelle registriert werden.

In der zusammengesetzten Datenbank müssen Sie jeden ODBC-Datenquellenserver definieren, auf den Sie zugreifen möchten.

#### Vorgehensweise:

Um eine Serverdefinition für eine ODBC-Datenquelle zu registrieren, gehen Sie wie folgt vor:

Setzen Sie die Anweisung `CREATE SERVER` ab.

Beispiel:

```
CREATE SERVER servername TYPE datenquellentyp
        VERSION version WRAPPER odbc
        OPTIONS (NODE 'knotenname')
```

Die Parameter `TYPE` und `VERSION` sind optional.

Obwohl `NODE` in der Anweisung `CREATE SERVER` als Option angegeben wird, ist dieser Wert für ODBC-Datenquellen erforderlich.

Verwenden Sie nach dem Erstellen der Serverdefinition die Anweisung `ALTER SERVER`, um Serveroptionen hinzuzufügen oder zu löschen.

Die nächste Task in dieser Tasksequenz ist das Erstellen einer Benutzerzuordnung für eine ODBC-Datenquelle.

### Zugehörige Tasks:

- „Erstellen einer Benutzerzuordnung für eine ODBC-Datenquelle“ auf Seite 354

### Zugehörige Referenzen:

- „ALTER SERVER statement“ in *SQL Reference, Volume 2*
- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641
- „Anweisung CREATE SERVER - Beispiele für den ODBC-Wrapper“ auf Seite 353

## Anweisung CREATE SERVER - Beispiele für den ODBC-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE SERVER verwendet wird, um Server für den ODBC-Wrapper zu registrieren. Dieser Abschnitt enthält ein vollständiges Beispiel, das zeigt, wie ein Server mit den erforderlichen Parametern registriert wird, sowie ein Beispiel mit zusätzlichen Serveroptionen.

### Vollständiges Beispiel:

Das folgende Beispiel zeigt, wie eine Serverdefinition für einen ODBC-Wrapper mit Hilfe der Anweisung CREATE SERVER registriert wird:

```
CREATE SERVER mysql_server TYPE mysql  
VERSION 4.0 WRAPPER odbc  
OPTIONS (NODE 'mysql_node', DBNAME 'venice')
```

#### *mysql\_server*

Ein Name, der dem ODBC-Datenquellenserver zugeordnet wird. Dieser Name muss eindeutig sein. Doppelte Servernamen sind nicht zulässig.

#### TYPE *mysql*

Gibt den Typ der Datenquelle an, für die der Zugriff konfiguriert wird. Dieser Parameter ist optional.

#### VERSION 4.0

Die Version der ODBC-Datenquelle, auf die Sie zugreifen wollen. Dieser Parameter ist optional.

#### WRAPPER *odbc*

Der Wrapper-Name, der in der Anweisung CREATE WRAPPER angegeben wurde.

#### NODE '*mysql\_node*'

Der Name des Knotens (der System-DSN-Name), der bei der Definition des DSN zur ODBC-Datenquelle zugeordnet wurde. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden. Unter Windows: Dieser Wert muss dem Namen eines System-DSN im Fenster für die ODBC-Datenverwaltung entsprechen. Unter UNIX: Die Dokumentation des Lieferanten des ODBC-Clients enthält Informationen über den zu verwendenden Wert.

Obwohl NODE in der Anweisung CREATE SERVER als Option angegeben wird, ist dieser Wert für ODBC-Datenquellen erforderlich.

#### DBNAME '*venice*'

Der Name der Datenbank, auf die zugegriffen werden soll. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

### Beispiel mit Serveroptionen:

Das folgende Beispiel zeigt, wie die Serveroptionen DB2\_TABLE\_QUOTE\_CHAR, DB2\_ID\_QUOTE\_CHAR und DB2\_AUTHID\_QUOTE\_CHAR verwendet werden:

Einige ODBC-Datenquellen (beispielsweise MySQL) können keine Anführungszeichen verarbeiten, in die Tabellennamen und Spaltennamen in SQL-Anweisungen eingeschlossen werden. Um auf diese Datenquellen zugreifen zu können, müssen Sie in der Anweisung CREATE SERVER die folgenden Serveroptionen angeben:

- DB2\_TABLE\_QUOTE\_CHAR ' ` '
- DB2\_ID\_QUOTE\_CHAR ' ` '
- DB2\_AUTHID\_QUOTE\_CHAR ' ` '

Das Zeichen ` ist der Begrenzer für Kennungen wie beispielsweise Schemanamen, Tabellennamen und Spaltennamen.

Beispiel:

```
CREATE SERVER mysql_server TYPE mysql
      VERSION 4.0 WRAPPER odbc
      OPTIONS (NODE 'mysql_node', DB2_TABLE_QUOTE_CHAR ' ` ',
              DB2_ID_QUOTE_CHAR ' ` ' DB2_AUTHID_QUOTE_CHAR ' ` ')
```

### Zugehörige Tasks:

- „Registrieren der Serverdefinitionen für eine ODBC-Datenquelle“ auf Seite 352

### Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

## Erstellen einer Benutzerzuordnung für eine ODBC-Datenquelle

Beim Hinzufügen von ODBC-Datenquellen zu Servern mit zusammengesetzten Datenbanken muss eine Benutzerzuordnung für eine ODBC-Datenquelle erstellt werden.

Wird versucht, auf eine ODBC-Datenquelle zuzugreifen, stellt der Server mit zusammengesetzten Datenbanken eine Verbindung zur Datenquelle her. Hierbei verwendet er eine Benutzer-ID und ein Kennwort, die für die betreffende Datenquelle gültig sind. Sie müssen eine Zuordnung (eine Benutzerzuordnung) zwischen jeder Benutzer-ID sowie jedem Kennwort des Servers mit zusammengesetzten Datenbanken und der entsprechenden Benutzer-ID und dem entsprechenden Kennwort der Datenquelle definieren. Erstellen Sie eine Benutzerzuordnung für jede Benutzer-ID, über die auf das System zusammengesetzter Datenbanken zugegriffen wird, um verteilte Anforderungen an die Informix-Datenquelle zu senden.

### Vorgehensweise:

Um der Benutzer-ID und dem Kennwort der ODBC-Datenquelle eine lokale Benutzer-ID zuzuordnen, setzen Sie die Anweisung CREATE USER MAPPING ab.

Beispiel:

```
CREATE USER MAPPING FOR benutzer-id SERVER servername
      OPTIONS (REMOTE_AUTHID 'ferne-id', REMOTE_PASSWORD 'fernes_kennwort')
```



Die nächste Task in dieser Tasksequenz ist das Testen der Verbindung zum ODBC-Datenquellenserver.

**Zugehörige Tasks:**

- „Testen der Verbindung zum ODBC-Datenquellenserver“ auf Seite 356

**Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE USER MAPPING - Beispiele für den ODBC-Wrapper“ auf Seite 355

## Anweisung CREATE USER MAPPING - Beispiele für den ODBC-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE USER MAPPING verwendet wird, um der Benutzer-ID und dem Kennwort einer ODBC-Datenquelle eine lokale Benutzer-ID zuzuordnen. Dieser Abschnitt enthält ein vollständiges Beispiel mit den erforderlichen Parametern sowie ein Beispiel, das zeigt, wie das DB2-Sonderregister USER mit der Anweisung CREATE USER MAPPING verwendet wird.

**Vollständiges Beispiel:**

Das folgende Beispiel zeigt, wie der Benutzer-ID und dem Kennwort einer ODBC-Datenquelle eine lokale Benutzer-ID zugeordnet wird:

```
CREATE USER MAPPING FOR arturo SERVER servername  
    OPTIONS (REMOTE_AUTHID 'art', REMOTE_PASSWORD 'red4blue')
```

*arturo* Gibt die lokale Benutzer-ID an, die Sie einer Benutzer-ID zuordnen, die in der ODBC-Datenquelle definiert wurde.

*servername*

Gibt den Namen der ODBC-Datenquelle an, den Sie in der Anweisung CREATE SERVER definiert haben.

*'art'* Gibt die Benutzer-ID in der ODBC-Datenquelle an, zu der Sie *arturo* zuordnen. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption FOLD\_ID in der Anweisung CREATE SERVER nicht auf 'U' oder 'L' setzen.

*'red4blue'*

Gibt das Kennwort an, das *'art'* zugeordnet ist. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption FOLD\_PW in der Anweisung CREATE SERVER nicht auf 'U' oder 'L' setzen.

**Beispiel für das Sonderregister:**

Das folgende Beispiel zeigt eine Anweisung CREATE USER MAPPING, die das Sonderregister USER enthält:

```
CREATE USER MAPPING FOR USER SERVER servername  
    OPTIONS (REMOTE_AUTHID 'art', REMOTE_PASSWORD 'red4blue')
```

Sie können das DB2-Sonderregister USER verwenden, um die Berechtigungs-ID der Person, die die Anweisung CREATE USER MAPPING absetzt, zur Berechtigungs-ID der Datenquelle zuzuordnen, die in der Benutzeroption REMOTE\_AUTHID angegeben wurde.



**Zugehörige Tasks:**

- „Erstellen einer Benutzerzuordnung für eine ODBC-Datenquelle“ auf Seite 354

**Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

## Testen der Verbindung zum ODBC-Datenquellenserver

Beim Hinzufügen von ODBC-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken muss die Verbindung zum ODBC-Datenquellenserver getestet werden.

Sie können die Verbindung zum ODBC-Datenquellenserver testen, indem Sie die Serverdefinition und Benutzerzuordnungen verwenden, die Sie definiert haben.

**Voraussetzungen:**

Die von Ihnen verwendete Datenquelle muss Durchgriffssitzungen unterstützen.

**Vorgehensweise:**

Um die Verbindung zu testen, gehen Sie wie folgt vor:

1. Öffnen Sie eine Durchgriffssitzung, um eine SQL-Anweisung SELECT für die Systemtabellen der ODBC-Datenquelle abzusetzen.

Beispiel:

```
SET PASSTHRU servername
SELECT COUNT(*) FROM schemaname.tabellenname
SET PASSTHRU RESET
```

Der *servername* ist der Name der ODBC-Datenquelle, die Sie in der Anweisung CREATE SERVER definiert haben.

Der *schemaname* ist der Name des Schemas in der fernen ODBC-Datenquelle. Wenn Ihre ODBC-Datenquelle keine Schemata unterstützt, übergehen Sie das Schema in der Anweisung.

Der *tabellenname* ist der Name der Tabelle in der fernen ODBC-Datenquelle.

Wenn die SQL-Anweisung SELECT eine Anzahl zurückgibt, sind Ihre Serverdefinition und Ihre Benutzerzuordnungen ordnungsgemäß eingerichtet.

2. Wenn die SQL-Anweisung SELECT einen Fehler zurückgibt, müssen Sie möglicherweise folgende Aktionen ausführen:
  - Stellen Sie sicher, dass die Datenquelle verfügbar ist.
  - Ist dies der Fall, prüfen Sie den Datenquellenserver, um sicherzustellen, dass er für ankommende Verbindungsanforderungen konfiguriert wurde.
  - Prüfen Sie Ihre Benutzerzuordnung, um sicherzustellen, dass die Einstellungen für die Optionen REMOTE\_AUTHID und REMOTE\_PASSWORD für Verbindungen zur ODBC-Datenquelle gültig sind. Ändern Sie gegebenenfalls die Benutzerzuordnung oder erstellen Sie gegebenenfalls eine andere Benutzerzuordnung.
  - Prüfen Sie den ODBC-Treiber auf dem DB2-Server mit zusammengeschlossenen Datenbanken, um sicherzustellen, dass er zum Herstellen einer Verbindung zum ODBC-Datenquellenserver ordnungsgemäß installiert und konfiguriert wurde. Unter Windows-Betriebssystemen: Verwenden Sie das Tool 'ODBC-Datenquellen-Administrator', um den Treiber zu prüfen. Unter UNIX-Betriebssystemen: Lesen Sie die entsprechende Dokumentation des Lieferanten des ODBC-Clients.

- Prüfen Sie Ihre Serverdefinition. Gegebenenfalls müssen Sie diese Definition löschen und erneut erstellen.

Die nächste Task in dieser Tasksequenz ist das Registrieren von Kurznamen für ODBC-Datenquellentabellen und -sichten.

#### Zugehörige Tasks:

- „Hinzufügen von ODBC-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 347
- „Vorbereiten des Servers mit zusammengeschlossenen Datenbanken für den Zugriff auf Datenquellen über ODBC“ auf Seite 349
- „Registrieren von Kurznamen für ODBC-Datenquellentabellen und -sichten“ auf Seite 357

#### Zugehörige Referenzen:

- „ALTER USER MAPPING statement“ in *SQL Reference, Volume 2*

## Registrieren von Kurznamen für ODBC-Datenquellentabellen und -sichten

Beim Hinzufügen von ODBC-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen Kurznamen für ODBC-Datenquellentabellen und -sichten registriert werden.

Registrieren Sie für jeden ODBC-Datenquellenserver, den Sie definieren, jeweils einen Kurznamen für alle Tabellen oder Sichten, auf die Sie zugreifen wollen. Verwenden Sie diese Kurznamen anstelle der Namen der Datenquellenobjekte, wenn Sie die ODBC-Datenquellen abfragen.

Angenommen, Sie definieren beispielsweise den Kurznamen *cust\_europe*, um eine Microsoft SQL Server-Tabelle mit dem Namen *italy* und dem Schemanamen *customers* darzustellen. Die SQL-Anweisung `SELECT * FROM cust_europe` vom Server mit zusammengeschlossenen Datenbanken ist zulässig. Die Anweisung `SELECT * FROM servername."customers"."italy"` ist jedoch nicht zulässig.

Zusätzlich zum Registrieren von Kurznamen für ODBC-Datenquellentabellen und -sichten können Sie auch Kurznamen für ferne Systemtabellen registrieren.

Wenn Ihre ODBC-Datenquelle keine Schemata unterstützt, übergehen Sie das Schema in der Anweisung `CREATE NICKNAME`.

Die zusammengeschlossene Datenbank verwendet Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Diese Statistiken werden gesammelt, wenn Sie unter Verwendung der Anweisung `CREATE NICKNAME` einen Kurznamen für ein Datenquellenobjekt registrieren. Die zusammengeschlossene Datenbank stellt das Vorhandensein des Objekts in der Datenquelle sicher und versucht dann, bestehende statistische Daten zur Datenquelle zu sammeln. Informationen, die für das Optimierungsprogramm nützlich sind, werden aus den Katalogen der Datenquelle gelesen und in den globalen Katalog auf dem Server mit zusammengeschlossenen Datenbanken aufgenommen. Da einige oder alle der Kataloginformationen von Datenquellen vom Optimierungsprogramm verwendet werden können, sollten Sie die Statistiken (unter Verwendung des Datenquellenbefehls, der dem DB2-Befehl **RUNSTATS** entspricht) in der Datenquelle aktualisieren, bevor Sie einen Kurznamen registrieren.

### Einschränkungen:

Wenn Sie einen Kurznamen für eine ODBC-Datenquelle erstellen, die Indizes enthält, zeichnet der ODBC-Wrapper die Indexinformationen nicht im Systemkatalog der zusammenschlossenen Datenbank auf. Sie müssen Indexspezifikationen für die Tabelle erstellen, indem Sie die Anweisung CREATE INDEX mit der Klausel SPECIFICATION ONLY verwenden.

### Vorgehensweise:

Um einen Kurznamen zu registrieren, setzen Sie die Anweisung CREATE NICKNAME ab.

Beispiel:

```
CREATE NICKNAME odbc_name FOR server_name."remote_schema"."remote.table"
```

Kurznamen können bis zu 128 Zeichen lang sein.

Wiederholen Sie diesen Schritt für jede ODBC-Tabelle oder -Sicht, für die Sie einen Kurznamen erstellen möchten.

Wenn Sie den Kurznamen erstellen, verwendet DB2 die Verbindung dazu, die Katalogtabellen der Datenquelle abzufragen. Diese Abfrage testet Ihre Verbindung zur ODBC-Datenquelle unter Verwendung des Kurznamens. Wenn die Verbindung nicht funktioniert, empfangen Sie eine Fehlermeldung.

### Zugehörige Tasks:

- „Erstellen von Indexspezifikationen für Datenquellenobjekte“ in *Systeme zusammenschlossener Datenbanken*

### Zugehörige Referenzen:

- „RUNSTATS Command“ in *Command Reference*
- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den ODBC-Wrapper“ auf Seite 358

## Anweisung CREATE NICKNAME - Beispiele für den ODBC-Wrapper

Dieser Abschnitt enthält ein Beispiel, das zeigt, wie die Anweisung CREATE NICKNAME verwendet wird, um einen Kurznamen für eine ODBC-Datenquellentabelle oder -sicht zu registrieren, auf die Sie zugreifen möchten.

Dieses Beispiel zeigt, wie ein fernes Objekt für die ODBC-Datenquelle angegeben wird, unter der der Kurzname zugeordnet ist:

```
CREATE NICKNAME cust_europe FOR servername."customers"."italy"
```

*cust\_europe*

Eindeutiger Kurzname für die Tabelle oder Sicht. Der Kurzname muss innerhalb des Schemas eindeutig sein.

**Hinweis:** Der Kurzname besteht aus zwei Teilen - dem Schema und dem Kurznamen. Wenn Sie beim Registrieren des Kurznamens das Schema auslassen, wird die Berechtigungs-ID des Benutzers, der den Kurznamen registriert, zum Schema des Kurznamens.

*servername."customers"."italy"*

Eine dreiteilige Kennung für das ferne Objekt.

- *servername* ist der Name, den Sie dem ODBC-Datenbankserver in der Anweisung CREATE SERVER zugeordnet haben.
- *customers* ist der Name des fernen Schemas, zu dem die Tabelle oder Sicht gehört. Wenn Ihre ODBC-Datenquelle keine Schemata unterstützt, übergehen Sie das Schema in der Anweisung CREATE NICKNAME.
- *italy* ist der Name der fernen Tabelle oder Sicht, auf die Sie zugreifen möchten.

Bei ODBC-Datenquellenobjekten muss unter Umständen die Groß-/Kleinschreibung beachtet werden. Schließen Sie sowohl den fernen Schemanamen als auch den fernen Tabellennamen in Anführungszeichen ein. Ansonsten setzt DB2 diese Namen in Großbuchstaben um.

#### **Zugehörige Tasks:**

- „Registrieren von Kurznamen für ODBC-Datenquellentabellen und -sichten“ auf Seite 357

#### **Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

---

## **Zugreifen auf Excel-Daten mit dem ODBC-Wrapper**

Sie können mit dem ODBC-Wrapper auf Microsoft Excel-Arbeitsmappen zugreifen, indem Sie den Excel ODBC-Treiber verwenden. Die Excel-Anwendung braucht nicht auf dem Server mit zusammengeschlossenen Datenbanken installiert zu sein. Der Excel ODBC-Treiber wird automatisch mit Microsoft Windows installiert.

Mit dem ODBC-Wrapper und dem Excel ODBC-Treiber können Sie auf Daten in allen Arbeitsblättern einer Arbeitsmappe zugreifen. Der Excel ODBC-Treiber interpretiert eine Arbeitsmappe als Datenbank und jedes Arbeitsblatt innerhalb der Arbeitsmappe als Tabelle.

Der Excel ODBC-Treiber unterstützt frühere Versionen von Excel-Arbeitsmappen, und zwar auch dann, wenn die Version der Excel-Anwendung, mit der die betreffenden Arbeitsmappen erstellt wurden, selbst nicht mehr unterstützt wird. Beispiel: Microsoft unterstützt keine Arbeitsblätter mehr, die mit Excel Version 4.0 erstellt wurden. Der Treiber unterstützt jedoch weiterhin Excel-Arbeitsblätter, die mit dieser Version erstellt wurden.

#### **Voraussetzungen:**

Der Excel ODBC-Treiber muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.

Der Server mit zusammengeschlossenen Datenbanken muss in der Lage sein, die Arbeitsblätter in der Excel-Arbeitsmappe zu öffnen und zu lesen, um Daten abrufen zu können. Daher müssen sich die Excel-Arbeitsmappen auf demselben Computer befinden wie der DB2-Server mit zusammengeschlossenen Datenbanken oder auf einem zugeordneten Netzlaufwerk, auf das Zugriff besteht.

### Einschränkungen:

- Ist eine Arbeitsmappe von einem Benutzer oder einer Anwendung bereits im Lese-/Schreibmodus geöffnet, kann der ODBC-Wrapper auf die Arbeitsblätter in der betreffenden Arbeitsmappe nicht zugreifen. Wenn der ODBC-Wrapper die Arbeitsmappe jedoch vor einem Benutzer oder einer Anwendung öffnet, kann der Benutzer bzw. die Anwendung die betreffende Arbeitsmappe dennoch im Lesemodus öffnen.
- Der Excel ODBC-Treiber erwartet, dass die erste belegte Zeile die Kennsätze für die Arbeitsblattspalten enthält. Sofern das Arbeitsblatt diese Kennsätze nicht enthält, müssen Sie eine Zeile mit den entsprechenden Spaltenkennsätzen in das Arbeitsblatt einfügen.
- Da der Excel ODBC-Treiber lediglich für Windows-Betriebssysteme zur Verfügung steht, können Sie den ODBC-Wrapper für den Zugriff auf Excel-Daten nur auf Servern mit zusammengeschlossenen Datenbanken verwenden, die ein Windows-Betriebssystem ausführen.
- Sie können für Excel-Arbeitsblätter Einfüge- und Aktualisierungsoperationen, jedoch keine Löschoperationen durchführen. Löschoperationen werden vom Excel ODBC-Treiber nicht unterstützt. Um Daten aus einem Arbeitsblatt zu löschen, müssen Sie das betreffende Arbeitsblatt in Excel öffnen und dann die erforderlichen Änderungen vornehmen.

### Vorgehensweise:

Um mit dem ODBC-Wrapper auf Excel-Arbeitsblätter zuzugreifen, gehen Sie wie folgt vor:

1. Stellen Sie sicher, dass die Excel-Arbeitsmappe, auf die Sie zugreifen wollen, sich entweder auf dem Server mit zusammengeschlossenen Datenbanken befindet oder auf einem zugeordneten Netzlaufwerk, auf das Zugriff besteht.
2. Methoden für den Zugriff auf Excel-Daten:
  - Ändern Sie gegebenenfalls das Layout der Daten in den Excel-Arbeitsblättern entsprechend den Anforderungen des Excel ODBC-Treibers.
  - Erstellen Sie gegebenenfalls die benannten Bereiche, auf die Sie zugreifen wollen.
3. Erstellen Sie einen System-DSN für die Arbeitsmappe, auf die Sie zugreifen wollen. Sie können den System-DSN mit Hilfe des ODBC-Datenquellen-Administrators konfigurieren. Der Name, der beim Erstellen des System-DSN angegeben wurde, wird als Wert für die Option `NODE` in der Anweisung `CREATE SERVER` zugeordnet.
4. Setzen Sie die Anweisung `CREATE WRAPPER` ab.  
Beispiel:  

```
CREATE WRAPPER odbc
```
5. Geben Sie die Speicherposition der Arbeitsmappe an, indem Sie ein Serverobjekt im Systemkatalog der zusammengeschlossenen Datenbank registrieren. Für den ODBC-Wrapper benötigen Sie für jeden DSN jeweils ein Serverobjekt. Der DSN wird der Arbeitsmappe zugeordnet, wenn der Excel ODBC-Treiber verwendet wird.

Um die Speicherposition der Arbeitsmappe anzugeben, setzen Sie die Anweisung `CREATE SERVER` ab und verwenden Sie den DSN als System-DSN für die Option `NODE`.

Beispiel:

```
CREATE SERVER compounds_workbook WRAPPER odbc  
OPTIONS (NODE 'compounds_workbook_dsn', PASSWORD 'n')
```

NODE *compounds\_workbook\_dsn* ist der System-DSN, den Sie erstellt haben. Die Optionen NODE und PASSWORD sind erforderlich, damit der ODBC-Wrapper auf Excel-Arbeitsblätter zugreifen kann.

Wiederholen Sie diesen Schritt für jede Arbeitsmappe, auf die Sie zugreifen wollen.

6. Setzen Sie die Anweisung CREATE NICKNAME ab, um einen Kurznamen für das Arbeitsblatt zu erstellen, auf das Sie zugreifen wollen. Die Syntax lautet wie folgt:

```
CREATE NICKNAME kurzname FOR servername.ferne_tabelle
```

- Wenn Sie für den Datenzugriff einen benannten Bereich erstellt haben, geben Sie den Namen des Bereichs im Abschnitt '*ferne\_tabelle*' der Anweisung CREATE NICKNAME an.

Beispiel: Ist der Name des Bereichs *testing*, lautet die Anweisung CREATE NICKNAME wie folgt:

```
CREATE NICKNAME compounds_nickname FOR compounds_workbook.testing
```

- Wenn Sie auf Daten im gesamten Arbeitsblatt und nicht nur in einem Bereich zugreifen wollen, geben Sie den Namen des Arbeitsblatts an, gefolgt vom Zeichen \$.

Beispiel: Ist der Name des Arbeitsblatts *Sheet1*, lautet die Anweisung CREATE NICKNAME wie folgt:

```
CREATE NICKNAME compounds_nick FOR compounds_workbook.Sheet1$
```

Wiederholen Sie diesen Schritt jeweils für alle Arbeitsblätter bzw. benannten Bereiche, auf die Sie zugreifen wollen.

## Ändern der standardmäßigen Datentypzuordnungen

Bei Verwendung des ODBC-Wrappers werden die unterstützten Datentypen vom Excel ODBC-Treiber festgelegt. Der Excel ODBC-Treiber ordnet die Excel-Datentypen den ODBC-Datentypen zu. Anschließend ordnet der ODBC-Wrapper die ODBC-Datentypen den DB2-Datentypen zu. Die DB2-Datentypen für die einzelnen Spalten werden in der Katalogtabelle der zusammengesetzten Datenbank gespeichert. Diese Datentypen werden als *lokale Datentypen* bezeichnet. Die folgende Tabelle enthält eine Liste der standardmäßigen Datentypzuordnungen:

Tabelle 79. Standardmäßige Datentypzuordnungen zwischen Excel und DB2 bei Verwendung des ODBC-Wrappers

Excel-Datentyp	ODBC-Datentyp	DB2-Datentyp
CURRENCY	SQL_NUMERIC	DECIMAL/DOUBLE
DATETIME	SQL_TIMESTAMP	TIMESTAMP
LOGICAL	SQL_BIT	SMALLINT
NUMBER	SQL_DOUBLE	DOUBLE
TEXT	SQL_VARCHAR	VARCHAR

Der ODBC-Wrapper unterstützt viele verschiedene Datentypen. Sie können die Standarddatentypen auch anderen DB2-Datentypen zuordnen. Um die lokalen Datentypen zu ändern, verwenden Sie die Anweisung ALTER NICKNAME. Die Liste der Datentypen, die über den ODBC-Wrapper verfügbar sind, ist sehr umfangreich und beinhaltet Datentypen wie beispielsweise LOBs und andere Doppelbyte-Datentypen.

| **Achtung:** Werden lokale Datentypen in andere Typen als in den ursprünglichen  
| Zuordnungen geändert, kann es möglicherweise zu Diskrepanzen bei den Daten-  
| typen kommen.

## | **Anweisung ALTER SERVER - Beispiele für den ODBC-Wrapper | für den Zugriff auf Excel-Daten**

| Über den Excel ODBC-Treiber ermöglicht der ODBC-Wrapper, dass Vergleichs-  
| elemente und Spaltenfunktionen im Pushdown-Modus zur Verarbeitung an die  
| Datenquelle gesendet werden. Der Treiber unterstützt auch Verknüpfungen zwi-  
| schen Arbeitsblättern. Durch das Senden der Verarbeitung im Pushdown-Modus an  
| die Datenquelle kann die Leistung verbessert werden, da sämtliche Datenquellen-  
| zeilen nicht an DB2 zurückgegeben werden, wenn in einer Abfrage auf ein Arbeits-  
| blatt verwiesen wird.

| Angenommen, Sie haben den Server *compounds\_workbook* definiert, um eine  
| Arbeitsmappe zu identifizieren, die Daten enthält, auf die Sie zugreifen wollen.  
| Wenn Sie für diese Arbeitsmappe nun angeben wollen, dass der Server mit zusam-  
| mengeschlossenen Datenbanken Operationen im Pushdown-Modus an den Excel  
| ODBC-Treiber zur Verarbeitung senden kann, lautet die entsprechende Anweisung  
| ALTER SERVER wie folgt:

```
| ALTER SERVER compounds_workbook  
|     OPTIONS (ADD PUSHDOWN 'Y')
```

| Um die Pushdown-Funktionalität nutzen zu können, müssen die folgenden ODBC-  
| Serveroptionen entweder in der Anweisung CREATE SERVER oder in der Anwei-  
| sung ALTER SERVER eingestellt werden:

- | • *PUSHDOWN 'Y'*
- | • *DB2\_GROUP\_BY 'Y'*
- | • *DB2\_COLFUNC 'Y'*

### | **Zugehörige Konzepte:**

- | • „Methoden für den Zugriff auf Excel-Daten“ auf Seite 30

### | **Zugehörige Tasks:**

- | • „Vorbereiten des Servers mit zusammengeschlossenen Datenbanken für den  
| Zugriff auf Datenquellen über ODBC“ auf Seite 349

### | **Zugehörige Referenzen:**

- | • „Auswählen des korrekten Wrappers“ auf Seite 29



---

## Optimieren der Konfiguration für ODBC-Datenquellen und Beheben von Konfigurationsfehlern

Nachdem Sie die Konfiguration für ODBC-Datenquellen eingerichtet haben, müssen Sie diese Konfiguration möglicherweise ändern, um die Leistung zu verbessern.

### Abrufen von ODBC-Traces

Wenn beim Zugriff auf die Datenquelle Probleme auftreten, können Sie ODBC-Traceinformationen abrufen, um die betreffenden Probleme zu analysieren und zu lösen. Das Aktivieren eines Trace wirkt sich jedoch auf die Systemleistung aus. Daher sollten Sie die Tracefunktion wieder inaktivieren, nachdem Sie die Probleme gelöst haben.

Auf Servern mit zusammengeschlossenen Datenbanken unter Windows klicken Sie **Trace durchführen** im Fenster 'ODBC-Datenquellen-Administrator' an.

Auf Servern mit zusammengeschlossenen Datenbanken unter UNIX: Lesen Sie die entsprechende Dokumentation des Lieferanten des ODBC-Clients.

#### Zugehörige Tasks:

- „Hinzufügen von ODBC-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 347

#### Zugehörige Referenzen:

- „db2set - DB2 Profile Registry Command“ in *Command Reference*





---

## Kapitel 18. Konfigurieren des Zugriffs auf OLE DB-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in OLE DB-Datenquellen gespeichert sind. Der Zugriff auf OLE DB-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält eine Liste der Tasks, die Sie ausführen müssen, und zeigt Beispiele zu den SQL-Anweisungen, die Sie benötigen, um den Server mit zusammengeschlossenen Datenbanken zu konfigurieren.

---

### Hinzufügen von OLE DB-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Microsoft OLE DB ist eine Gruppe von OLE-/COM-Schnittstellen, die für Anwendungen einheitlichen Zugriff auf Daten bereitstellen, die in verschiedenen Informationsquellen gespeichert sind. Die Komponenten-DBMS-Architektur von OLE DB definiert OLE DB-Consumer und OLE DB-Provider. Ein OLE DB-Consumer ist ein beliebiges System bzw. eine beliebige Anwendung, das/die OLE DB-Schnittstellen verwendet. Ein OLE DB-Provider ist eine Komponente, die OLE DB-Schnittstellen verfügbar macht.

Die OLE DB-Oberfläche ermöglicht den Zugriff auf OLE DB-Provider, die Microsoft OLE DB 2.0 (oder höher) entsprechen.

Der OLE DB-Wrapper wird zum Erstellen von Tabellenfunktionen verwendet. Mit diesem Wrapper können keine Kurznamen für Datenquellentabellen- und -sichten erstellt werden.

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf OLE DB-Datenquellen zu konfigurieren, müssen Sie dem Server Informationen zu den OLE DB-Providern zur Verfügung stellen.

Der Zugriff auf die OLE DB-Datenquellen kann über die DB2-Befehlszentrale oder die DB2-Befehlszeile konfiguriert werden.

#### **Voraussetzungen:**

- DB2 Information Integrator muss auf einem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss auf dem Server mit zusammengeschlossenen Datenbanken vorhanden sein.

#### **Einschränkung:**

Der OLE DB-Wrapper dient lediglich als Hilfe beim Registrieren benutzerdefinierter externer OLE DB-Tabellenfunktionen. Im Gegensatz zu anderen Wrappern verwendet der OLE DB-Wrapper für den Zugriff auf Daten, die in Datenquellen gespeichert sind, keine Kurznamen.

### Vorgehensweise:

Um OLE DB-Datenquellen einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen, gehen Sie wie folgt vor:

1. Registrieren Sie den Wrapper.
2. Registrieren Sie die Serverdefinition.
3. Erstellen Sie die Benutzerzuordnungen.

Nach der Konfiguration des Zugriffs auf die OLE DB-Datenquelle verwenden Sie die Anweisung `CREATE FUNCTION`, um eine benutzerdefinierte externe OLE DB-Tabellenfunktion in der zusammengeschlossenen Datenbank zu registrieren.

### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzübersicht“ auf Seite 63

### Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Registrieren des OLE DB-Wrappers“ auf Seite 366

---

## Registrieren des OLE DB-Wrappers

Beim Hinzufügen von OLE DB-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken muss der OLE DB-Wrapper registriert werden.

Das Registrieren eines Wrappers ist für den Zugriff auf OLE DB-Datenquellen erforderlich. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken verwendet, um mit Datenquellen zu kommunizieren und Daten aus Datenquellen abzurufen. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

### Vorgehensweise:

Um einen Wrapper zu registrieren, setzen Sie die Anweisung `CREATE WRAPPER` ab und geben Sie den Standardnamen für den Wrapper an.

Beispiel:

```
CREATE WRAPPER OLEDB
```

**Empfehlung:** Verwenden Sie den standardmäßigen Wrappernamen `OLEDB`. Wenn Sie den Wrapper mit dem Standardnamen registrieren, übernimmt der Server mit zusammengeschlossenen Datenbanken automatisch den standardmäßigen Bibliotheksnamen, der diesem Wrapper zugeordnet ist.

Sollte zwischen dem Namen des Wrappers und einem bereits vorhandenen Wrappernamen in der zusammengeschlossenen Datenbank ein Namenskonflikt entstehen, können Sie den Standardnamen für den Wrapper durch einen beliebigen Namen ersetzen. In diesem Fall müssen Sie in der Anweisung `CREATE WRAPPER` den Parameter `LIBRARY` angeben.

Um beispielsweise einen Wrapper mit dem Namen `oledb_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken zu registrieren, der das Betriebssystem Windows verwendet, setzen Sie die folgende Anweisung ab:

```
CREATE WRAPPER oledb_wrapper LIBRARY 'db2oledb.dll'
```

Der Name der von Ihnen angegebene Wrapperbibliotheksdatei hängt von dem Betriebssystem auf dem Server mit zusammengeschlossenen Datenbanken ab. Den korrekten Namen, der in der Anweisung `CREATE WRAPPER` angegeben werden muss, finden Sie in der Liste der OLE DB-Wrapper-Bibliotheksdateien.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den OLE DB-Wrapper.

**Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine OLE DB-Datenquelle“ auf Seite 367

**Zugehörige Referenzen:**

- „OLE DB-Wrapper-Bibliotheksdateien“ auf Seite 367
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

---

## OLE DB-Wrapper-Bibliotheksdateien

Die folgende Tabelle enthält eine Liste der Verzeichnispfade und Bibliotheksdateinamen für den OLE DB-Wrapper.

Wenn Sie DB2 Information Integrator installieren, wird die Bibliotheksdatei dem in der Tabelle aufgeführten Verzeichnispfad hinzugefügt.

Wenn Sie einen Wrapper registrieren, geben Sie nur den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 80. Speicherpositionen und Dateinamen der OLE DB-Wrapperbibliothek*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
Windows	%DB2PATH%\bin	db2oledb.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows lautet `C:\Program Files\IBM\SQLLIB`.

**Zugehörige Tasks:**

- „Registrieren des OLE DB-Wrappers“ auf Seite 366

---

## Registrieren der Serverdefinitionen für eine OLE DB-Datenquelle

Beim Hinzufügen von OLE DB-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen die Serverdefinitionen für eine OLE DB-Datenquelle registriert werden.

In der zusammengeschlossenen Datenbank müssen Sie jeden OLE DB-Datenquellenserver definieren, auf den Sie zugreifen möchten.

### Vorgehensweise:

Um eine Serverdefinition für eine OLE DB-Datenquelle zu registrieren, gehen Sie wie folgt vor. Setzen Sie die Anweisung CREATE SERVER ab.

Beispiel:

```
CREATE SERVER servername WRAPPER OLEDB
    OPTIONS (CONNECTSTRING 'Provider=Microsoft.Jet.OLEDB.4.0;
                        Data Source=c:\msdasdk\bin\oledb\nwind.mdb')
```

Die nächste Task in dieser Tasksequenz ist das Erstellen einer Benutzerzuordnung für eine OLE DB-Datenquelle.

### Zugehörige Tasks:

- „Erstellen einer Benutzerzuordnung für eine OLE DB-Datenquelle“ auf Seite 369

### Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE SERVER - Beispiele für den OLE DB-Wrapper“ auf Seite 368

---

## Anweisung CREATE SERVER - Beispiele für den OLE DB-Wrapper

Dieser Abschnitt enthält ein Beispiel, das zeigt, wie die Anweisung CREATE SERVER verwendet wird, um Server für den OLE DB-Wrapper zu registrieren.

Das folgende Beispiel zeigt eine Anweisung CREATE SERVER:

```
CREATE SERVER nwind WRAPPER OLEDB
    OPTIONS (CONNECTSTRING 'Provider=Microsoft.Jet.OLEDB.4.0;
                        Data Source=c:\msdasdk\bin\oledb\nwind.mdb',
            COLLATING_SEQUENCE 'Y')
```

**nwind** Ein Name, der der OLE DB-Datenquelle zugeordnet wird. Dieser Name muss eindeutig sein. Doppelte Servernamen sind nicht zulässig.

#### WRAPPER OLEDB

Der Wrapper-Name, der in der Anweisung CREATE WRAPPER angegeben wurde.

**CONNECTSTRING** 'Provider=Microsoft.Jet.OLEDB.4.0; Data Source=c:\msdasdk\bin\oledb\nwind.mdb'

Stellt Initialisierungseigenschaften bereit, die für das Herstellen einer Verbindung zu einer Datenquelle erforderlich sind. Die Zeichenfolge enthält eine Reihe von Paaren aus Schlüsselwort und Wert, die jeweils durch ein Semikolon getrennt werden. Das Gleichheitszeichen (=) trennt jedes Schlüsselwort vom zugehörigen Wert. Schlüsselwörter sind die Beschreibungen der OLE DB-Initialisierungseigenschaften (Eigenschaftsmenge DBPROPSET\_DBINT) oder providerspezifische Schlüsselwörter.

Die vollständige Syntax und Semantik der Option CONNECTSTRING finden Sie im Handbuch *Microsoft OLE DB 2.0 Programmer's Reference and Data Access SDK* Microsoft Press 1998.

#### COLLATING\_SEQUENCE 'Y'

Gibt an, ob die Datenquelle dieselbe Sortierfolge verwendet wie die Sortierfolge von DB2 für UNIX und Windows.

Gültige Werte sind 'Y' (die Datenquelle verwendet die Sortierfolge von DB2 für UNIX und Windows) und 'N' (die Datenquelle verwendet eine andere Sortierfolge als die von DB2 für UNIX und Windows). Der Standardwert ist 'N'.

**Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine OLE DB-Datenquelle“ auf Seite 367

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

---

## Erstellen einer Benutzerzuordnung für eine OLE DB-Datenquelle

Beim Hinzufügen von OLE-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken muss eine Benutzerzuordnung für eine OLE-Datenquelle erstellt werden.

Wird versucht, auf eine OLE-Datenquelle zuzugreifen, stellt der Server mit zusammengeschlossenen Datenbanken eine Verbindung zur Datenquelle her. Hierbei verwendet er eine Benutzer-ID und ein Kennwort, die für die betreffende Datenquelle gültig sind. Sie müssen eine Zuordnung (eine Benutzerzuordnung) zwischen jeder Benutzer-ID sowie jedem Kennwort des Servers mit zusammengeschlossenen Datenbanken und der entsprechenden Benutzer-ID und dem entsprechenden Kennwort der Datenquelle definieren. Erstellen Sie eine Benutzerzuordnung für jede Benutzer-ID, über die auf das System zusammengeschlossener Datenbanken zugegriffen wird, um verteilte Anforderungen an die Informix-Datenquelle zu senden.

**Vorgehensweise:**

Um der Benutzer-ID und dem Kennwort der OLE-Datenquelle eine lokale Benutzer-ID zuzuordnen, setzen Sie die Anweisung CREATE USER MAPPING ab.

Beispiel:

```
CREATE USER MAPPING FOR benutzer-id SERVER servername
    OPTIONS (REMOTE_AUTHID 'ferne-id', REMOTE_PASSWORD 'fernes_kennwort')
```

Ist entweder das Kennwort für die OLE DB-Datenquelle oder das Kennwort für den Server mit zusammengeschlossenen Datenbanken kürzer als acht Zeichen, schlagen SQL-Anweisungen, die auf die OLE DB-Datenquelle zugreifen, fehl. Die folgende Fehlermeldung wird angezeigt:

```
SQL30082N Die Verbindung konnte auf
Grund der Sicherheitsbedingung "15" ("PROCESSING FAILURE") nicht hergestellt werden.
SQLSTATE=08001
```

Um dieses Problem zu vermeiden, müssen Sie entweder das Kennwort für die OLE DB-Datenquelle oder das Kennwort für den Server mit zusammengeschlossenen Datenbanken auf eine Länge von mindestens acht Zeichen ändern.

**Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE USER MAPPING - Beispiele für den OLE DB-Wrapper“ auf Seite 370

---

## Anweisung CREATE USER MAPPING - Beispiele für den OLE DB-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE USER MAPPING verwendet wird, um der Benutzer-ID und dem Kennwort einer OLE-Datenquelle eine lokale Benutzer-ID zuzuordnen. Dieser Abschnitt enthält ein vollständiges Beispiel mit den erforderlichen Parametern sowie ein Beispiel, das zeigt, wie das DB2-Sonderregister USER mit der Anweisung CREATE USER MAPPING verwendet wird.

### Vollständiges Beispiel:

Das folgende Beispiel zeigt, wie der Benutzer-ID und dem Kennwort einer OLE-Datenquelle eine lokale Benutzer-ID zugeordnet wird:

```
CREATE USER MAPPING FOR laura SERVER Nwind
  OPTIONS (REMOTE_AUTHID 'lulu', REMOTE_PASSWORD 'raiders')
```

*laura* Die lokale Benutzer-ID, die Sie einer Benutzer-ID zuordnen, die in der OLE DB-Datenquelle definiert wurde.

### SERVER *Nwind*

Der Name des OLE DB-Servers, den Sie in der Anweisung CREATE SERVER definiert haben.

### REMOTE\_AUTHID '*lulu*'

Die Benutzer-ID auf dem OLE DB-Server, zu der Sie *laura* zuordnen. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

### REMOTE\_PASSWORD '*raiders*'

Das Kennwort, das '*lulu*' zugeordnet ist. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

### Beispiel für das Sonderregister:

Das folgende Beispiel zeigt eine Anweisung CREATE USER MAPPING, die das Sonderregister USER enthält:

```
CREATE USER MAPPING FOR USER SERVER Nwind
  OPTIONS (REMOTE_AUTHID 'lulu', REMOTE_PASSWORD 'raiders')
```

Sie können das DB2-Sonderregister USER verwenden, um die Berechtigungs-ID der Person, die die Anweisung CREATE USER MAPPING absetzt, zur Berechtigungs-ID der Datenquelle zuzuordnen, die in der Benutzeroption REMOTE-AUTHID angegeben wurde.

### Zugehörige Tasks:

- „Erstellen einer Benutzerzuordnung für eine OLE DB-Datenquelle“ auf Seite 369

### Zugehörige Referenzen:

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

---

## Kapitel 19. Konfigurieren des Zugriffs auf Oracle-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Oracle-Datenquellen gespeichert sind. Der Zugriff auf Oracle-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält eine Liste der Tasks, die Sie ausführen müssen, wenn Sie den Server mit zusammengeschlossenen Datenbanken konfigurieren, zeigt Beispiele zu den benötigten SQL-Anweisungen und bietet Informationen zur Optimierung und Fehlerbehebung beim Konfigurieren des Servers mit zusammengeschlossenen Datenbanken.

---

### Hinzufügen von Oracle zu einem System zusammengeschlossener Datenbanken

#### Hinzufügen von Oracle-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Oracle-Datenquellen zu konfigurieren, müssen Sie dem Server Informationen zu den Datenquellen und -objekten zur Verfügung stellen, auf die Sie zugreifen möchten.

Der Zugriff des Servers mit zusammengeschlossenen Datenbanken auf Oracle-Datenquellen kann über die DB2-Steuerzentrale oder die DB2-Befehlszeile konfiguriert werden. Die DB2-Steuerzentrale umfasst einen Assistenten, der Sie durch die erforderlichen Schritte zum Konfigurieren des Servers mit zusammengeschlossenen Datenbanken führt.

#### Voraussetzungen:

- Die Oracle-Client-Software muss auf dem Server mit zusammengeschlossenen Datenbanken installiert und konfiguriert sein.
- DB2 Information Integrator muss auf einem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss auf dem Server mit zusammengeschlossenen Datenbanken vorhanden sein.

#### Vorgehensweise:

Um Oracle-Datenquellen einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen, gehen Sie wie folgt vor:

1. Richten Sie die Oracle-Clientkonfigurationsdatei ein und testen Sie diese Datei.
2. Setzen Sie die Oracle-Umgebungsvariablen.
3. Registrieren Sie den Wrapper.
4. Registrieren Sie die Serverdefinition.
5. Erstellen Sie die Benutzerzuordnungen.
6. Testen Sie die Verbindung zum Oracle-Server.
7. Registrieren Sie Kurznamen für Oracle-Tabellen, -Sichten und -Synonyme.



### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

### Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Einrichten und Testen der Oracle-Clientkonfigurationsdatei“ auf Seite 376
- „Registrieren des Oracle-Wrappers“ auf Seite 377
- „Registrieren der Serverdefinitionen für eine Oracle-Datenquelle“ auf Seite 378
- „Erstellen der Benutzerzuordnungen für eine Oracle-Datenquelle“ auf Seite 381
- „Testen der Verbindung zum Oracle-Server“ auf Seite 382
- „Registrieren von Kurznamen für Oracle-Tabellen und -Sichten“ auf Seite 384
- „Optimierung und Fehlerbehebung Konfiguration für Oracle-Datenquellen“ auf Seite 386
- „Einrichten der Oracle-Umgebungsvariablen“ auf Seite 372
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43

### Zugehörige Referenzen:

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## Einrichten der Oracle-Umgebungsvariablen

Beim Hinzufügen von Oracle zu Servern mit zusammengeschlossenen Datenbanken müssen die Oracle-Umgebungsvariablen eingerichtet werden.

Wenn Sie DB2 Information Integrator installieren, wird während des Installationsprozesses versucht, die Oracle-Umgebungsvariablen in der Datei `db2dj.ini` einzurichten.

In folgenden Fällen werden die Umgebungsvariablen in der Datei `db2dj.ini` nicht eingerichtet:

- Die Oracle-Client-Software wird nach Einrichten des DB2-Servers mit zusammengeschlossenen Datenbanken installiert.
- Die Oracle-Client-Software wurde nicht installiert.

Die folgenden Umgebungsvariablen sind für Oracle gültig:

- `ORACLE_HOME`
- `ORACLE_BASE` (optional)
- `ORA_NLS` (optional)
- `TNS_ADMIN` (optional)
- `NLS_LANG` (optional)

Die optionalen Umgebungsvariablen müssen manuell eingerichtet werden.

### Voraussetzungen:

Ein Server mit zusammengeschlossenen Datenbanken, der ordnungsgemäß für den Zugriff auf Datenquelle eingerichtet ist. Dies umfasst die Installation und Konfiguration der erforderlichen Software wie beispielsweise der Client-Software von Datenquellen.

### Einschränkungen:

Siehe Abschnitt: Einschränkungen für die Datei db2dj.ini

### Vorgehensweise:

Um die erforderlichen Umgebungsvariablen automatisch einzurichten, gehen Sie wie folgt vor:

1. Installieren und konfigurieren Sie die Client-Software auf dem DB2-Server mit zusammengeschlossenen Datenbanken, sofern die Software nicht bereits installiert wurde.
2. Richten Sie die erforderlichen Umgebungsvariablen ein. Sie können die Umgebungsvariablen automatisch einrichten, indem Sie die Installation von DB2 Information Integrator erneut ausführen. Klicken Sie über die Klickstartleiste **Produkte installieren** an und befolgen Sie die Anweisungen des Assistenten.

Um die Oracle-Umgebungsvariablen manuell einzurichten, gehen Sie wie folgt vor:

1. Editieren Sie die Datei db2dj.ini.
  - Auf Servern mit zusammengeschlossenen Datenbanken unter Windows befindet sich diese Datei im Verzeichnis sqllib\cfg.
  - Auf Servern mit zusammengeschlossenen Datenbanken unter UNIX befindet sich diese Datei im Verzeichnis sqllib/cfg.

Die Datei db2dj.ini enthält Konfigurationsdaten zu der auf dem Server mit zusammengeschlossenen Datenbanken installierten Oracle-Client-Software. Falls die Datei nicht vorhanden ist, können Sie mit Hilfe eines beliebigen Texteditors eine neue Datei mit dem Namen db2dj.ini erstellen. In der Datei db2dj.ini müssen Sie den vollständig qualifizierten Pfad für die Variablen angeben, da ansonsten Fehler auftreten.

2. Richten Sie je nach Bedarf die folgenden Umgebungsvariablen ein:

#### ORACLE\_HOME

Setzen Sie die Umgebungsvariable ORACLE\_HOME auf den Pfad des Verzeichnisses, in dem die Oracle-Client-Software installiert ist. Geben Sie den vollständig qualifizierten Pfad für die Variable an: ORACLE\_HOME *oracle-ausgangsverzeichnis*. Ist das Oracle-Ausgangsverzeichnis beispielsweise '\usr\oracle\8.1.7' lautet der Eintrag in der Datei db2dj.ini folgendermaßen: ORACLE\_HOME=\usr\oracle\8.1.7

**Hinweis:** Falls ein einzelner Benutzer des Exemplars zusammengeschlossener Datenbanken die Umgebungsvariable ORACLE\_HOME gesetzt hat, wird diese Einstellung vom Exemplar zusammengeschlossener Datenbanken nicht verwendet. Das Exemplar zusammengeschlossener Datenbanken verwendet ausschließlich den von Ihnen in der Datei db2dj.ini gesetzten Wert für ORACLE\_HOME.

## ORACLE\_BASE

Die Umgebungsvariable ORACLE\_BASE gibt die Basis (Root) der Baumstruktur des Oracle-Clientverzeichnisses an. Wenn Sie die Variable ORACLE\_BASE beim Installieren der Oracle-Client-Software gesetzt haben, setzen Sie die Umgebungsvariable ORACLE\_BASE auf dem Server mit zusammengeschlossenen Datenbanken. Beispiel:

```
ORACLE_BASE=oracle-rootverzeichnis
```

## ORA\_NLS\*

Wenn Ihr System mehrere Versionen von Oracle verwendet, müssen Sie Folgendes sicherstellen:

- Die entsprechende Variable ORA\_NLS wurde gesetzt.
- Die entsprechenden NLS-Datendateien für die verwendeten Versionen sind verfügbar.

Die standortspezifischen Daten werden in einem Verzeichnis gespeichert, das von der Umgebungsvariable ORA\_NLS\* angegeben wird. Für jede neue Version von Oracle ist ein unterschiedliches ORA\_NLS-Datenverzeichnis vorhanden.

*Tabelle 81. Die Oracle-Variablen ORA\_NLS zur Angabe der Speicherposition der für die Ländereinstellungen (Locale) spezifischen Daten nach Versionen.*

Oracle-Version	Umgebungsvariable
7.2	ORA_NLS
7.3	ORA_NLS32
8.x, 9.x	ORA_NLS33
10g	ORA_NLS10

Für UNIX-Server mit zusammengeschlossenen Datenbanken, die auf Datenquellen von Oracle 8.1 zugreifen, setzen Sie die Umgebungsvariable ORA\_NLS33 beispielsweise wie folgt:

```
ORA_NLS33=oracle-ausgangsverzeichnis/ocommon/nls/admin/<data>
```

## TNS\_ADMIN

- Auf Servern mit zusammengeschlossenen Datenbanken unter Windows sucht der Oracle-Client nach der Datei tnsnames.ora im Verzeichnis %ORACLE\_HOME%\NETWORK\ADMIN, wobei %ORACLE\_HOME% in der Datei db2dj.ini definiert ist. Ist die Datei tnsnames.ora nicht im Verzeichnis %ORACLE\_HOME%\NETWORK\ADMIN vorhanden, müssen Sie die Umgebungsvariable TNS\_ADMIN auf dem Server mit zusammengeschlossenen Datenbanken setzen.
- Auf Servern mit zusammengeschlossenen Datenbanken unter AIX, Linux oder HP-UX sucht der Client nach der Datei tnsnames.ora im Verzeichnis /etc. Kann der Client die Datei tnsnames.ora im Verzeichnis /etc nicht finden, sucht er im Verzeichnis \$ORACLE\_HOME/network/admin nach dieser Datei, wobei \$ORACLE\_HOME in der Datei db2dj.ini definiert ist. Kann der Client die Datei tnsnames.ora nicht finden, müssen Sie die Umgebungsvariable TNS\_ADMIN setzen. Setzen Sie die Variable in der Datei db2dj.ini auf den Pfad, in dem sich die Datei tnsnames.ora befindet.

Befindet sich die Datei tnsnames.ora beispielsweise im Verzeichnis /home/oracle, setzen Sie die Variable wie folgt:

```
TNS_ADMIN=/home/oracle
```

- Auf Servern mit zusammengeschlossenen Datenbanken unter Solaris sucht der Client nach der Datei `tnsnames.ora` im Verzeichnis `/var/opt/oracle`. Kann der Client die Datei `tnsnames.ora` im Verzeichnis `/var/opt/oracle` nicht finden, sucht er im Verzeichnis `$ORACLE_HOME/network/admin` nach dieser Datei, wobei `$ORACLE_HOME` in der Datei `db2dj.ini` definiert ist. Kann der Client die Datei `tnsnames.ora` nicht finden, müssen Sie die Umgebungsvariable `TNS_ADMIN` setzen. Setzen Sie die Variable in der Datei `db2dj.ini` auf den Pfad, in dem sich die Datei `tnsnames.ora` befindet.

Befindet sich die Datei `tnsnames.ora` beispielsweise im Verzeichnis `/home/oracle`, setzen Sie die Variable wie folgt:

```
TNS_ADMIN=/home/oracle
```

3. Unter UNIX müssen Sie die Datei `.profile` des DB2-Exemplars mit der Oracle-Umgebungsvariable aktualisieren. Setzen Sie dazu den folgenden Befehl ab:

```
export ORACLE_HOME=oracle-ausgangsverzeichnis
export PATH=$ORACLE_HOME/bin:$PATH
```

Hierbei ist `oracle-ausgangsverzeichnis` das Verzeichnis, in dem die Oracle-Client-Software installiert ist.

4. Unter UNIX müssen Sie das DB2-Exemplar `.profile` ausführen, indem Sie Folgendes eingeben:

```
. $HOME/.profile
```

5. Um sicherzustellen, dass die Umgebungsvariablen auf dem Server mit zusammengeschlossenen Datenbanken gesetzt werden, müssen Sie das DB2-Exemplar stoppen und erneut starten: Setzen Sie die folgenden Befehle ab, um das DB2-Exemplar zu stoppen und erneut zu starten:

```
db2stop
db2start
```

### Einrichten der Konvertierung der Oracle-Codepage:

Für die Konvertierung der Oracle-Codepage können Sie die optionale Umgebungsvariable `NLS_LANG` setzen.

Jedes Mal, wenn der Oracle-Wrapper eine Verbindung zu einer Oracle-Datenquelle herstellt, ermittelt der Wrapper, welcher Codepagewert für die betreffende Verbindung verwendet werden soll. Wird die Umgebungsvariable `NLS_LANG` in der Datei `db2dj.ini` auf dem Server mit zusammengeschlossenen Datenbanken gesetzt, verwendet der Wrapper den Wert in der Datei `db2dj.ini`. Die Datei `db2dj.ini` enthält Konfigurationsdaten zu der auf dem Server mit zusammengeschlossenen Datenbanken installierten Oracle-Client-Software.

Wird die Variable `NLS_LANG` auf dem Server mit zusammengeschlossenen Datenbanken nicht gesetzt, ermittelt der Wrapper das Gebiet und die Codepage der zusammengeschlossenen Datenbank. Der Wrapper setzt die Variable `NLS_LANG` auf die am ehesten zutreffende Oracle-Ländereinstellung (Locale). Treffen keine Ländereinstellungen zu, wird `NLS_LANG` auf den Wert `'American_America.US7ASCII'` gesetzt.

Wenn Sie auf eine Datenquelle mit Daten zugreifen, für die die chinesische Codepage GB 18030 verwendet wird, muss Ihre zusammengeschlossene Datenbank die Codepage UTF-8 verwenden. Für Oracle-Datenquellen setzt der Oracle-Wrapper die Oracle-Umgebungsvariable `NLS_LANG` auf folgenden Wert:

```
NLS_LANG=Simplified_Chinese_China.UTF8
```

Wenn Sie den Oracle 9i-Client verwenden, ändern Sie die Einstellung für NLS\_ \_LANG in der Datei db2dj.ini auf den Wert 'Simplified Chinese\_China. AL32UTF8', damit der Oracle 9i-Client die GB 18030-Daten ordnungsgemäß in Unicode umsetzt. Beispiel:

```
NLS_LANG=Simplified Chinese_China.AL32UTF8
```

Die Dokumentation zur jeweils verwendeten Oracle-Software enthält eine Liste der gültigen Ländereinstellungen (Locales).

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für Oracle-Tabellen und -Sichten“ auf Seite 384

#### Zugehörige Referenzen:

- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68
- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68

## Einrichten und Testen der Oracle-Clientkonfigurationsdatei

Beim Hinzufügen von Oracle-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken muss die Oracle-Clientkonfigurationsdatei eingerichtet und getestet werden. Die Clientkonfigurationsdatei wird zum Herstellen einer Verbindung zu Oracle-Datenbanken eingesetzt, wobei die Clientbibliotheken verwendet werden, die auf dem Server mit zusammengeschlossenen Datenbanken installiert sind. Diese Datei gibt die Speicherposition aller Oracle-Datenbankserver und den Typ der Verbindung (Protokoll) für den Datenbankserver an. Der Standardname für die Oracle-Clientkonfigurationsdatei ist tnsnames.ora.

#### Vorgehensweise:

Um die Oracle-Clientkonfigurationsdatei einzurichten und zu testen, gehen Sie wie folgt vor:

1. Verwenden Sie das Oracle-Konfigurationsdienstprogramm NET8/NET, das im Lieferumfang der Oracle-Client-Software enthalten ist. Weitere Informationen zur Verwendung dieses Dienstprogramms sind in der Installationsdokumentation von Oracle enthalten. In der Datei tnsnames.ora ist SID (oder SERVICE\_NAME) der Name des Oracle-Exemplars und HOST der Name des Hosts, auf dem sich der Oracle-Server befindet. In welchem Verzeichnis die Datei tnsnames.ora erstellt wird, hängt vom Betriebssystem Ihres Servers mit zusammengeschlossenen Datenbanken ab.
  - Unter UNIX-Betriebssystemen lautet der Standardpfad und -name dieser Datei \$ORACLE\_HOME/network/admin.
  - Unter Windows-Betriebssystemen lautet der Standardpfad und -name dieser Datei %ORACLE\_HOME%\NETWORK\ADMIN.
2. Wenn Sie die Datei tnsnames.ora in einem anderen Pfad als dem Standardsuchpfad speichern möchten, müssen Sie die Umgebungsvariable TNS\_ADMIN so setzen, dass sie die gewünschte Speicherposition der Datei angibt.
  - a. Editieren Sie die Datei db2dj.ini, die im Verzeichnis sql11ib/cfg gespeichert ist, und setzen Sie die Umgebungsvariable TNS\_ADMIN wie folgt:

```
TNS_ADMIN=x:/path/
```
  - b. Setzen Sie die folgenden Befehle ab, um das DB2-Exemplar erneut zu starten und sich zu vergewissern, dass die Umgebungsvariable im Programm definiert ist:

```
db2stop
db2start
```

3. Verwenden Sie das Oracle-Tool **sqlplus**, um die Verbindung zu testen und so sicherzustellen, dass die Client-Software eine Verbindung zum Oracle-Server herstellen kann.

Die nächste Task in dieser Tasksequenz ist das Registrieren des Oracle-Wrappers.

**Zugehörige Tasks:**

- „Registrieren des Oracle-Wrappers“ auf Seite 377
- „Optimierung und Fehlerbehebung Konfiguration für Oracle-Datenquellen“ auf Seite 386

## Registrieren des Oracle-Wrappers

Beim Hinzufügen von Oracle-Datenquellen zu einem Server mit zusammengesetzten Datenbanken muss der Oracle-Wrapper registriert werden.

Das Registrieren eines Wrappers ist für den Zugriff auf Oracle-Datenquellen erforderlich. Wrapper werden von Servern mit zusammengesetzten Datenbanken verwendet, um mit Datenquellen zu kommunizieren und Daten aus Datenquellen abzurufen. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

**Vorgehensweise:**

Um einen Wrapper zu registrieren, setzen Sie die Anweisung `CREATE WRAPPER` ab und geben Sie den Standardnamen für den Wrapper an.

Beispiel:

```
CREATE WRAPPER NET8
```

**Empfehlung:** Verwenden Sie den standardmäßigen Wrappernamen `NET8`. Wenn Sie den Wrapper durch Verwendung eines der beiden Standardnamen registrieren, übernimmt der Server mit zusammengesetzten Datenbanken automatisch den standardmäßigen Bibliotheksnamen, der diesem Wrapper zugeordnet ist.

Wenn der Wrapper-Name zu Konflikten mit einem bestehenden Wrapper-Namen in der zusammengesetzten Datenbank führt, können Sie den standardmäßigen Wrapper-Namen durch einen Namen Ihrer Wahl ersetzen. Wenn Sie einen Namen verwenden, der sich von einem der Standardnamen unterscheidet, müssen Sie in der Anweisung `CREATE WRAPPER` den Parameter `LIBRARY` mit angeben. Um beispielsweise einen Wrapper mit dem Namen `oracle_wrapper` auf dem Server mit zusammengesetzten Datenbanken zu registrieren, der das Betriebssystem `AIX` verwendet, setzen Sie die folgende Anweisung ab:

```
CREATE WRAPPER oracle_wrapper LIBRARY 'libdb2net8.a'
```

Der Name der von Ihnen angegebene Wrapperbibliotheksdatei hängt von dem Betriebssystem auf dem Server mit zusammengesetzten Datenbanken ab. Den korrekten Namen, der in der Anweisung `CREATE WRAPPER` angegeben werden muss, finden Sie in der Liste der Oracle-Wrapper-Bibliotheksdateien.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den Oracle-Wrapper.

**Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine Oracle-Datenquelle“ auf Seite 378



#### Zugehörige Referenzen:

- „Oracle-Wrapper-Bibliotheksdateien“ auf Seite 378
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Oracle-Wrapper-Bibliotheksdateien

Die folgende Tabelle enthält eine Liste der Verzeichnispfade und Bibliotheksdateinamen für den Oracle-Wrapper.

Wenn Sie DB2 Information Integrator installieren, werden dem in der Tabelle aufgeführten Verzeichnispfad 3 Bibliotheksdateien hinzugefügt. Wird der Server mit zusammengesetzten Datenbanken beispielsweise unter AIX ausgeführt, werden die Wrapperbibliotheksdateien `libdb2net8.a`, `libdb2net8F.a` und `libdb2net8U.a` dem Verzeichnispfad hinzugefügt.

Wenn Sie einen Wrapper registrieren, geben Sie nur den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 82. Speicherpositionen und Dateinamen der Oracle-Wrapperbibliothek*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	<code>/usr/opt/db2_08_01/lib/</code>	<code>libdb2net8F.a (NET8)</code>
HP-UX	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2net8F.sl (NET8)</code>
Linux	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2net8F.so (NET8)</code>
Solaris	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2net8F.so (NET8)</code>
Windows	<code>%DB2PATH%\bin</code>	<code>db2net8.dll (NET8)</code>

`%DB2PATH%` ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows lautet `C:\Program Files\IBM\SQLLIB`.

#### Zugehörige Tasks:

- „Registrieren der Serverdefinitionen für eine Oracle-Datenquelle“ auf Seite 378

#### Zugehörige Referenzen:

- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Registrieren der Serverdefinitionen für eine Oracle-Datenquelle

Beim Hinzufügen von Oracle-Datenquellen zu Servern mit zusammengesetzten Datenbanken müssen die Serverdefinitionen für eine Oracle-Datenquelle registriert werden.

In der zusammengesetzten Datenbank müssen Sie jeden Oracle-Server definieren, auf den Sie zugreifen möchten. Zunächst müssen Sie den Knotennamen der Oracle-Datenquelle suchen und diesen Knotennamen anschließend bei der Registrierung des Servers verwenden.

### Vorgehensweise:

Um eine Serverdefinition für eine Oracle-Datenquelle zu registrieren, gehen Sie wie folgt vor:

1. Suchen Sie den Knotennamen in der Oracle-Datei `tnsnames.ora`.

#### Beispiel für die Datei 'tnsnames.ora':

```
paris_node =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = somehost)(PORT = 1521)))
    (CONNECT_DATA = (SERVICE_NAME = ora9i.seel)))
```

In diesem Beispiel ist `paris_node` der Knotenwert, der in der Anweisung `CREATE SERVER` zu verwenden ist.

Obwohl der Parameter *knotenname* in der SQL-Anweisung `CREATE SERVER` als Option angegeben wird, ist er für Oracle-Datenquellen erforderlich.

2. Setzen Sie die Anweisung `CREATE SERVER` ab.

Beispiel:

```
CREATE SERVER servername TYPE oracle VERSION 8.1.7 WRAPPER net8
  OPTIONS (NODE 'knotenname')
```

Verwenden Sie nach dem Erstellen der Serverdefinition die Anweisung `ALTER SERVER`, um Serveroptionen hinzuzufügen oder zu löschen.

Die nächste Task in dieser Tasksequenz ist das Erstellen der Benutzerzuordnungen für eine Oracle-Datenquelle.

### Zugehörige Tasks:

- „Erstellen der Benutzerzuordnungen für eine Oracle-Datenquelle“ auf Seite 381

### Zugehörige Referenzen:

- „ALTER SERVER statement“ in *SQL Reference, Volume 2*
- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641
- „Anweisung CREATE SERVER - Beispiele für den Oracle-Wrapper“ auf Seite 379

## Anweisung CREATE SERVER - Beispiele für den Oracle-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung `CREATE SERVER` verwendet wird, um Server für den Oracle-Wrapper zu registrieren. Dieser Abschnitt enthält ein vollständiges Beispiel, das zeigt, wie ein Server mit den erforderlichen Parametern registriert wird, sowie ein Beispiel mit zusätzlichen Serveroptionen.

### Vollständiges Beispiel:

Das folgende Beispiel zeigt, wie eine Serverdefinition für einen Oracle-Wrapper mit Hilfe der Anweisung `CREATE SERVER` registriert wird:

```
CREATE SERVER oraserver TYPE oracle VERSION 8.1.7 WRAPPER net8
  OPTIONS (NODE 'paris_node')
```



*oraserver*

Ein Name, den Sie dem Oracle-Datenbankserver zuordnen. Dieser Name muss eindeutig sein. Doppelte Servernamen sind nicht zulässig.

**TYPE** *oracle*

Gibt den Typ des Datenquellenservers an, für den Sie den Zugriff konfigurieren. Der Parameter TYPE für die Wrapper SQLNET und NET8 muss den Wert *oracle* haben.

**VERSION** *8.1.7*

Die Version des Oracle-Datenbankservers, auf den Sie zugreifen möchten. Die Oracle-Versionen 7.3.4, 8.x und 9.x werden unterstützt.

**WRAPPER** *net8*

Der Name, den Sie in der Anweisung CREATE WRAPPER angegeben haben.

**NODE** *'paris\_node'*

Der Name des Knotens, in dem sich der Oracle-Datenbankserver befindet. Den Knotennamen finden Sie in der Datei *tnsnames.ora*.

Obwohl der Knotenname in der Anweisung CREATE SERVER als Option angegeben wird, ist er für Oracle-Datenquellen erforderlich.

### **Beispiel mit Serveroptionen:**

Beim Erstellen der Serverdefinition können Sie in der Anweisung CREATE SERVER zusätzliche Serveroptionen angeben. Diese Serveroptionen umfassen allgemeine Serveroptionen sowie Serveroptionen, die für Oracle spezifisch sind.

DB2 geht davon aus, dass alle VARCHAR-Spalten in Oracle folgende Leerzeichen enthalten. Wenn Sie sich sicher sind, dass alle VARCHAR-Spalten in der Oracle-Datenbank keine folgenden Leerzeichen enthalten, können Sie eine Serveroption setzen, mit der angegeben wird, dass die Datenquelle eine VARCHAR-Vergleichssemantik ohne Leerzeichen als Füllzeichen verwendet.

Das folgende Beispiel zeigt eine Oracle-Serverdefinition mit dieser Serveroptionen:

```
CREATE SERVER oraserver TYPE oracle VERSION 8.1.7 WRAPPER net8  
OPTIONS (NODE 'paris_node', VARCHAR_NO_TRAILING_BLANKS 'Y')
```

Verwenden Sie die Serveroption VARCHAR\_NO\_TRAILING\_BLANKS, wenn keine der Spalten folgende Leerzeichen enthält. Wenn nur einige der VARCHAR-Spalten folgende Leerzeichen enthalten, können Sie mit Hilfe der Anweisung ALTER NICKNAME eine Option für diese spezifischen Spalten setzen.

### **Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine Oracle-Datenquelle“ auf Seite 378

### **Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

## Erstellen der Benutzerzuordnungen für eine Oracle-Datenquelle

Beim Hinzufügen von Oracle-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen die Benutzerzuordnungen für eine Oracle-Datenquelle erstellt werden.

Wird versucht, auf einen Oracle-Server zuzugreifen, stellt der Server mit zusammengeschlossenen Datenbanken eine Verbindung zur Datenquelle her. Hierbei verwendet er eine Benutzer-ID und ein Kennwort, die für die betreffende Datenquelle gültig sind. Sie müssen eine Zuordnung (eine Benutzerzuordnung) zwischen jeder Benutzer-ID sowie jedem Kennwort des Servers mit zusammengeschlossenen Datenbanken und der entsprechenden Benutzer-ID und dem entsprechenden Kennwort der Datenquelle definieren. Erstellen Sie eine Benutzerzuordnung für jede Benutzer-ID, über die auf das System zusammengeschlossener Datenbanken zugegriffen wird, um verteilte Anforderungen an die Oracle-Datenquelle zu senden.

### Vorgehensweise:

Um der Benutzer-ID und dem Kennwort des Oracle-Servers eine lokale Benutzer-ID zuzuordnen, setzen Sie die Anweisung CREATE USER MAPPING ab.

Beispiel:

```
CREATE USER MAPPING FOR benutzer-id SERVER oraserver
      OPTIONS (REMOTE_AUTHID 'ferne-id', REMOTE_PASSWORD 'fernes_kennwort')
```

Die nächste Task in dieser Tasksequenz ist das Testen der Verbindung zum Oracle-Server.

### Zugehörige Tasks:

- „Testen der Verbindung zum Oracle-Server“ auf Seite 382

### Zugehörige Referenzen:

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE USER MAPPING - Beispiele für den Oracle-Wrapper“ auf Seite 381

## Anweisung CREATE USER MAPPING - Beispiele für den Oracle-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE USER MAPPING verwendet wird, um die Benutzer-ID eines Servers mit zusammengeschlossenen Datenbanken einer Benutzer-ID und einem Kennwort eines Oracle-Servers zuzuordnen. Dieser Abschnitt enthält ein vollständiges Beispiel mit den erforderlichen Parametern sowie ein Beispiel, das zeigt, wie das DB2-Sonderregister USER mit der Anweisung CREATE USER MAPPING verwendet wird.

### Vollständiges Beispiel:

Das folgende Beispiel zeigt, wie die Benutzer-ID eines Servers mit zusammengeschlossenen Datenbanken der Benutzer-ID und dem Kennwort eines Oracle-Servers zugeordnet wird:

```
CREATE USER MAPPING FOR robert SERVER oraserver
  OPTIONS (REMOTE_AUTHID 'rob', REMOTE_PASSWORD 'then4now')
```

*robert* Gibt die lokale Benutzer-ID an, die Sie einer Benutzer-ID zuordnen, die auf einem Oracle-Server definiert wurde.

**SERVER** *oraserver*

Gibt den Namen des Oracle-Servers an, den Sie in der Anweisung CREATE SERVER definiert haben.

**REMOTE\_AUTHID** '*rob*'

Gibt die Benutzer-ID auf dem Oracle-Datenbankserver an, zu der Sie *robert* zuordnen. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption FOLD\_ID in der Anweisung CREATE SERVER nicht auf 'U' oder 'L' setzen.

**REMOTE\_PASSWORD** '*then4now*'

Gibt das Kennwort an, das '*rob*' zugeordnet ist. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption FOLD\_PW in der Anweisung CREATE SERVER nicht auf 'U' oder 'L' setzen.

**Beispiel für das Sonderregister:**

Das folgende Beispiel zeigt eine Anweisung CREATE USER MAPPING, die das Sonderregister USER enthält:

```
CREATE USER MAPPING FOR USER SERVER oraserver
  OPTIONS (REMOTE_AUTHID 'rob', REMOTE_PASSWORD 'then4now')
```

Sie können das DB2-Sonderregister USER verwenden, um die Berechtigungs-ID der Person, die die Anweisung CREATE USER MAPPING absetzt, zur Berechtigungs-ID der Datenquelle zuzuordnen, die in der Benutzeroption REMOTE\_AUTHID angegeben wurde.

**Einschränkung:** Die Benutzer-ID in der Oracle-Datenquelle muss unter Verwendung des Oracle-Befehls **create user** (Benutzer erstellen) mit der Klausel 'identified by' (identifiziert durch) statt der Klausel 'identified externally' (extern identifiziert) erstellt worden sein.

**Zugehörige Tasks:**

- „Erstellen der Benutzerzuordnungen für eine Oracle-Datenquelle“ auf Seite 381

**Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

## Testen der Verbindung zum Oracle-Server

Beim Hinzufügen von Oracle-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken muss die Verbindung zum Oracle-Server getestet werden.

Sie können die Verbindung zum Oracle-Server testen, indem Sie die Serverdefinition und Benutzerzuordnungen verwenden, die Sie definiert haben.

### Vorgehensweise:

Um die Verbindung zu testen, gehen Sie wie folgt vor:

1. Öffnen Sie eine Durchgriffssitzung, um eine SQL-Anweisung SELECT für die Oracle-Systemtabellen abzusetzen.

Beispiel:

```
SET PASSTHRU name_des_fernen_servers
SELECT count(*) FROM sys.all_tables
SET PASSTHRU RESET
```

Wenn die SQL-Anweisung SELECT eine Anzahl zurückgibt, sind Ihre Serverdefinition und Ihre Benutzerzuordnung ordnungsgemäß eingerichtet.

2. Wenn die SQL-Anweisung SELECT einen Fehler zurückgibt, müssen Sie möglicherweise folgende Aktionen ausführen:
  - Den Oracle-Server prüfen, um sicherzustellen, dass er für ankommende Verbindungsanforderungen konfiguriert wurde.
  - Ihre Benutzerzuordnung prüfen, um sicherzustellen, dass die Einstellungen für die Optionen REMOTE\_AUTHID und REMOTE\_PASSWORD für Verbindungen zum Oracle-Server gültig sind. Ändern Sie gegebenenfalls die Benutzerzuordnung oder erstellen Sie gegebenenfalls eine andere Benutzerzuordnung.
  - Die Oracle-Client-Software auf dem DB2-Server mit zusammengeschlossenen Datenbanken prüfen, um sicherzustellen, dass sie für das Herstellen einer Verbindung zum Oracle-Server ordnungsgemäß installiert und konfiguriert wurde.
  - Die Einstellungen Ihrer DB2-Variablen für zusammengeschlossene Datenbanken prüfen, um sicherzustellen, dass sie für das Arbeiten mit dem Oracle-Server korrekt definiert sind. Zu diesen Variablen gehören die Systemumgebungsvariablen, die db2dj.ini-Variablen und die Variable der DB2-Profilregistrierdatenbank (db2set).
  - Prüfen Ihrer Serverdefinition. Gegebenenfalls müssen Sie diese Definition löschen und erneut erstellen.

Die nächste Task in dieser Tasksequenz ist das Registrieren von Kurznamen für Oracle-Tabellen und -Sichten.

### Zugehörige Tasks:

- „Hinzufügen von Oracle-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 371
- „Einrichten und Testen der Oracle-Clientkonfigurationsdatei“ auf Seite 376
- „Registrieren von Kurznamen für Oracle-Tabellen und -Sichten“ auf Seite 384
- „Einrichten der Oracle-Umgebungsvariablen“ auf Seite 372

### Zugehörige Referenzen:

- „ALTER USER MAPPING statement“ in *SQL Reference, Volume 2*

## Registrieren von Kurznamen für Oracle-Tabellen und -Sichten

Beim Hinzufügen von Oracle-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen Kurznamen für Oracle-Tabellen und -Sichten registriert werden.

Registrieren Sie für jeden Oracle-Server, den Sie definieren, jeweils einen Kurznamen für alle Tabellen oder Sichten, auf die Sie zugreifen wollen. Verwenden Sie diese Kurznamen anstelle der Namen der Datenquellenobjekte, wenn Sie die Oracle-Server abfragen.

Die zusammengeschlossene Datenbank verwendet Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Diese Statistiken werden gesammelt, wenn Sie unter Verwendung der Anweisung CREATE NICKNAME einen Kurznamen für ein Datenquellenobjekt registrieren. Die zusammengeschlossene Datenbank stellt das Vorhandensein des Objekts in der Datenquelle sicher und versucht dann, bestehende statistische Daten zur Datenquelle zu sammeln. Informationen, die für das Optimierungsprogramm nützlich sind, werden aus den Katalogen der Datenquelle gelesen und in den globalen Katalog auf dem Server mit zusammengeschlossenen Datenbanken aufgenommen. Da einige oder alle der Kataloginformationen von Datenquellen vom Optimierungsprogramm verwendet werden können, sollten Sie die Statistiken (unter Verwendung des Datenquellenbefehls, der dem DB2-Befehl RUNSTATS entspricht) in der Datenquelle aktualisieren, bevor Sie einen Kurznamen registrieren.

### Vorgehensweise:

Um einen Kurznamen zu registrieren, setzen Sie die Anweisung CREATE NICKNAME ab.

Beispiel:

```
CREATE NICKNAME oracle_name FOR oraserver."remote_schema"."remote.table"
```

Kurznamen können bis zu 128 Zeichen lang sein.

Wiederholen Sie diesen Schritt für jede Oracle-Tabelle oder -Sicht, für die Sie einen Kurznamen erstellen möchten.

Wenn Sie den Kurznamen erstellen, verwendet DB2 die Verbindung dazu, den Datenquellenkatalog abzufragen. Diese Abfrage testet Ihre Verbindung zur Datenquelle unter Verwendung des Kurznamens. Wenn die Verbindung nicht funktioniert, empfangen Sie eine Fehlermeldung.

### Zugehörige Referenzen:

- „RUNSTATS Command“ in *Command Reference*
- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den Oracle-Wrapper“ auf Seite 385

## Anweisung CREATE NICKNAME - Beispiele für den Oracle-Wrapper

Dieser Abschnitt enthält ein Beispiel, das zeigt, wie die Anweisung CREATE NICKNAME verwendet wird, um einen Kurznamen für eine Oracle-Tabelle oder -Sicht zu registrieren, auf die Sie zugreifen möchten.

Dieses Beispiel zeigt, wie ein fernes Objekt für den Oracle-Server angegeben wird, unter dem der Kurzname zugeordnet ist:

```
CREATE NICKNAME PARISINV FOR oraserver."france"."inventory"
```

*PARISINV*

Ein eindeutiger Kurzname, der zur Kennzeichnung der Oracle-Tabelle oder -Sicht verwendet wird.

**Hinweis:** Der Kurzname besteht aus zwei Teilen - dem Schema und dem Kurznamen. Wenn Sie bei der Registrierung des Kurznamens das Schema auslassen, wird die Berechtigungs-ID des Benutzers, der den Kurznamen registriert, zum Schema des Kurznamens.

*oraserver."france"."inventory"*

Eine dreiteilige Kennung für das ferne Objekt:

- *oraserver* ist der Name, den Sie dem Oracle-Datenbankserver in der Anweisung CREATE SERVER zugeordnet haben.
- *france* ist der Name des fernen Schemas, zu dem die Tabelle oder Sicht gehört.
- *inventory* ist der Name der fernen Tabelle oder Sicht, auf die Sie zugreifen möchten.

Der Server mit zusammengeschlossenen Datenbanken setzt die Namen der Oracle-Schemata und -Tabellen in Großbuchstaben um, sofern Sie die Namen nicht in Anführungszeichen setzen.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für Oracle-Tabellen und -Sichten“ auf Seite 384

### Zugehörige Referenzen:

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

---

## Optimierung und Fehlerbehebung Konfiguration für Oracle-Datenquellen

Nachdem Sie die Konfiguration für Oracle-Datenquellen eingerichtet haben, können Sie diese Konfiguration ändern, um die Leistung zu verbessern.

### Konnektivitätsprobleme

Für jeden Parameter HOST im Abschnitt DESCRIPTION der Datei `tnsnames.ora` müssen Sie möglicherweise die TCP/IP-Datei `hosts` aktualisieren. Ob Sie diese Datei aktualisieren, hängt davon ab, wie TCP/IP in Ihrem Netzwerk konfiguriert wurde. Ein Teil des Netzwerks muss den fernen Hostnamen, der im Abschnitt DESCRIPTION in der Datei `tnsnames.ora` angegeben wurde, in eine Adresse umsetzen.

Wenn Ihr Netzwerk über einen benannten Server verfügt, der den Hostnamen erkennt, müssen Sie die TCP/IP-Datei `hosts` nicht aktualisieren. Andernfalls benötigen Sie für den fernen Host einen Eintrag. Wenden Sie sich an Ihren Netzwerkadministrator, um zu ermitteln, wie Ihr Netzwerk konfiguriert ist. Wenn Sie die Datei `hosts` aktualisieren müssen, hängt die Dateiadresse vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken ab:

#### **Auf Servern mit zusammengeschlossenen Datenbanken unter UNIX**

Aktualisieren Sie die Datei `/etc/hosts`.

#### **Auf Servern mit zusammengeschlossenen Datenbanken unter Windows**

Aktualisieren Sie die Datei `x:\winnt\system32\drivers\etc\hosts`.

#### **Zugehörige Tasks:**

- „Hinzufügen von Oracle-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 371

#### **Zugehörige Referenzen:**

- „db2set - DB2 Profile Registry Command“ in *Command Reference*

---

## Kapitel 20. Konfigurieren des Zugriffs auf Sybase-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Sybase-Datenquellen gespeichert sind. Der Zugriff auf Sybase-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält eine Liste der Tasks, die Sie ausführen müssen, wenn Sie den Server mit zusammengeschlossenen Datenbanken konfigurieren, zeigt Beispiele zu den benötigten SQL-Anweisungen und bietet Informationen zur Optimierung und Fehlerbehebung beim Konfigurieren des Servers mit zusammengeschlossenen Datenbanken.

---

### Hinzufügen von Sybase zu einem System zusammengeschlossener Datenbanken

#### Hinzufügen von Sybase-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Sybase-Datenquellen zu konfigurieren, müssen Sie dem Server Informationen zu den Datenquellen und -objekten zur Verfügung stellen, auf die Sie zugreifen möchten.

Der Zugriff des Servers mit zusammengeschlossenen Datenbanken auf Sybase-Datenquellen kann über die DB2-Steuerzentrale oder die DB2-Befehlszeile konfiguriert werden. Die DB2-Steuerzentrale umfasst einen Assistenten, der Sie durch die erforderlichen Schritte zum Konfigurieren des Servers mit zusammengeschlossenen Datenbanken führt.

#### Voraussetzungen:

- DB2 Information Integrator muss auf einem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss auf dem Server mit zusammengeschlossenen Datenbanken vorhanden sein.

#### Vorgehensweise:

Um Sybase-Datenquellen einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen, gehen Sie wie folgt vor:

1. Richten Sie die Sybase-Clientkonfigurationsdatei ein und testen Sie diese Datei.
2. Setzen Sie die Sybase-Umgebungsvariablen.
3. Registrieren Sie den Wrapper.
4. Registrieren Sie die Serverdefinition.
5. Erstellen Sie die Benutzerzuordnungen.
6. Testen Sie die Verbindung zum Sybase-Server.
7. Registrieren Sie Kurznamen für Sybase-Tabellen und -Sichten.



### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

### Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43
- „Einrichten und Testen der Sybase-Clientkonfigurationsdatei“ auf Seite 391
- „Registrieren des Sybase-Wrappers“ auf Seite 392
- „Registrieren der Serverdefinitionen für eine Sybase-Datenquelle“ auf Seite 394
- „Erstellen einer Benutzerzuordnung für eine Sybase-Datenquelle“ auf Seite 397
- „Testen der Verbindung zum Sybase-Server“ auf Seite 399
- „Registrieren von Kurznamen für Sybase-Tabellen und -Sichten“ auf Seite 400
- „Optimierung und Fehlerbehebung Konfiguration für Sybase-Datenquellen“ auf Seite 402
- „Einstellen der Sybase-Umgebungsvariablen“ auf Seite 388

### Zugehörige Referenzen:

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## Einstellen der Sybase-Umgebungsvariablen

Beim Hinzufügen von Sybase zu einem Server mit zusammengeschlossenen Datenbanken müssen die Sybase-Umgebungsvariablen eingerichtet werden.

Wenn Sie DB2 Information Integrator installieren, wird während des Installationsprozesses versucht, die Sybase-Umgebungsvariablen in der Datei `db2dj.ini` einzurichten.

In folgenden Fällen werden die Umgebungsvariablen in der Datei `db2dj.ini` nicht eingerichtet:

- Die Sybase-Client-Software wird nach Einrichten des Servers mit zusammengeschlossenen Datenbanken installiert.
- Die Sybase-Client-Software wurde nicht installiert.

Die folgenden Umgebungsvariablen sind für Sybase gültig:

- SYBASE
- SYBASE\_OCS (erforderlich für Sybase Version 12 oder höher)
- SYBASE\_CHARSET (optional)

Die optionale Umgebungsvariable muss manuell eingerichtet werden.

### Voraussetzungen:

Ein Server mit zusammengeschlossenen Datenbanken, der ordnungsgemäß für den Zugriff auf Datenquellen eingerichtet ist. Dies umfasst die Installation und Konfiguration der erforderlichen Software wie beispielsweise der Client-Software von Datenquellen.

### Einschränkungen:

Siehe Abschnitt: Einschränkungen für die Datei db2dj.ini

### Vorgehensweise:

Um die erforderlichen Umgebungsvariablen automatisch einzurichten, gehen Sie wie folgt vor:

1. Installieren und konfigurieren Sie die Client-Software auf dem Server mit zusammengeschlossenen Datenbanken, sofern die Software nicht bereits installiert wurde.
2. Richten Sie die erforderlichen Umgebungsvariablen ein. Sie können die Umgebungsvariablen automatisch einrichten, indem Sie die Installation von DB2 Information Integrator erneut ausführen. Klicken Sie über die Klickstartleiste **Produkte installieren** an und befolgen Sie die Anweisungen des Assistenten.

Um die Umgebungsvariablen manuell einzurichten, gehen Sie wie folgt vor:

1. Editieren Sie die Datei db2dj.ini.
  - Auf Servern mit zusammengeschlossenen Datenbanken unter Windows befindet sich diese Datei im Verzeichnis sql11ib\cfg.
  - Auf Servern mit zusammengeschlossenen Datenbanken unter UNIX befindet sich diese Datei im Verzeichnis sql11ib/cfg.

Die Datei db2dj.ini enthält Konfigurationsdaten zu der auf dem Server mit zusammengeschlossenen Datenbanken installierten Sybase-Client-Software. Falls die Datei nicht vorhanden ist, können Sie mit Hilfe eines beliebigen Texteditors eine neue Datei mit dem Namen db2dj.ini erstellen. In der Datei db2dj.ini müssen Sie den vollständig qualifizierten Pfad für die Variablen angeben, da ansonsten Fehler auftreten.

2. Richten Sie je nach Bedarf die folgenden Umgebungsvariablen ein:

#### SYBASE

Setzen Sie die Umgebungsvariable SYBASE auf den Pfad des Verzeichnisses, in dem die Sybase Open Client-Software installiert ist. Geben Sie für diese Variable den vollständig qualifizierten Pfad an. Beispiel:

```
SYBASE=sybase-ausgangsverzeichnis
```

Lautet der Verzeichnispfad beispielsweise D:/djxc1ient/sybase/V12/OCS-12\_5, geben Sie die Variable SYBASE wie folgt an:

```
SYBASE=D:/djxc1ient/sybase/V12
```

## SYBASE\_OCS

Bei Sybase Open Client Version 12 oder höher setzen Sie die Umgebungsvariable SYBASE\_OCS auf den Namen des OCS-Verzeichnisses. Geben Sie hierbei nicht den vollständig qualifizierten Pfad an. Die Umgebungsvariable SYBASE\_OCS gibt die Version und das Release der installierten Sybase Open Client-Software an.

```
SYBASE_OCS=OCS-version_release
```

Lautet der Verzeichnispfad beispielsweise

D:/djxc1ient/sybase/V12/OCS-12\_5, geben Sie die Variable SYBASE\_OCS wie folgt an:

```
SYBASE_OCS=OCS-12_5
```

## SYBASE\_CHARSET

Setzen Sie die Variable SYBASE\_CHARSET auf den Namen des Zeichensatzes, den Sie verwenden wollen. Der Sybase-Wrapper ermittelt anhand der Variablen SYBASE\_CHARSET, welcher Zeichensatz verwendet werden soll. Wird die Variable SYBASE\_CHARSET nicht gesetzt, verwendet der Wrapper den Sybase-Zeichensatz, der mit dem Zeichensatz übereinstimmt, der für die Codepage des Servers mit zusammengeschlossenen Datenbanken angegeben wurde. Ist kein übereinstimmender Sybase-Zeichensatz vorhanden, verwendet der Wrapper den Zeichensatz 'iso\_1'. Das Verzeichnis \$SYBASE\charsets enthält eine Liste der Namen von gültigen Zeichensätzen. Beispiel:

```
SYBASE_CHARSET=iso_1
```

Hierbei ist iso\_1 der Name des Zeichensatzes, der verwendet werden soll.

3. Aktualisieren Sie die Datei .profile des Exemplars der zusammengeschlossenen Datenbank mit der Umgebungsvariablen SYBASE. Setzen Sie hierzu die folgenden Befehle ab:

```
export  
SYBASE=<sybase-ausgangsverzeichnis  
export SYBASE_OCS=OCS-version_release  
export PATH=$SYBASE/bin:$PATH
```

4. Führen Sie über das Ausgangsverzeichnis die Datei .profile des Exemplars der zusammengeschlossenen Datenbank aus, indem Sie Folgendes eingeben:

```
. .profile
```

5. Um sicherzustellen, dass die Umgebungsvariablen auf dem Server mit zusammengeschlossenen Datenbanken gesetzt werden, müssen Sie das Exemplar der zusammengeschlossenen Datenbank stoppen und erneut starten. Setzen Sie hierzu die folgenden Befehle ab:

```
db2stop  
db2start
```

Die nächste Task in dieser Tasksequenz ist das Registrieren des Sybase-Wrappers.

### Zugehörige Tasks:

- „Registrieren von Kurznamen für Sybase-Tabellen und -Sichten“ auf Seite 400

### Zugehörige Referenzen:

- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68
- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68

## Einrichten und Testen der Sybase-Clientkonfigurationsdatei

Beim Hinzufügen von Sybase-Datenquellen zu Servern mit zusammengesetzten Datenbanken muss die Sybase-Clientkonfigurationsdatei eingerichtet und getestet werden.

Die Clientkonfigurationsdatei wird zum Herstellen einer Verbindung zu Sybase eingesetzt. Dabei werden die Bibliotheken von Sybase Open Client verwendet, die auf dem Server mit zusammengesetzten Datenbanken installiert sind. Diese Datei gibt die Speicherposition aller Exemplare von Sybase SQL Server und Adaptive Server Enterprise und den Typ der Verbindung (Protokoll) für den Datenbankserver an.

Für alle Exemplare des Servers mit zusammengesetzten Datenbanken, die mit Sybase verbunden werden sollen, muss jeweils eine Clientkonfigurationsdatei eingerichtet werden. Die Schritte, die zum Einrichten und Testen dieser Datei ausgeführt werden müssen, hängen vom Betriebssystem ab, das auf dem Server mit zusammengesetzten Datenbanken ausgeführt wird.

### Vorgehensweise:

Um die Clientkonfigurationsdatei einzurichten und zu testen, gehen Sie wie folgt vor:

#### Unter UNIX-Betriebssystemen:

1. Verwenden Sie zum Einrichten der Clientkonfigurationsdatei das Dienstprogramm, das im Lieferumfang der Software von Sybase Open Client enthalten ist. Diese Datei wird im Verzeichnis \$SYBASE erstellt. Der Name der Datei lautet `interfaces`. Weitere Informationen zur Verwendung dieses Dienstprogramms sind in der Sybase-Dokumentation enthalten.
2. Stellen Sie dem Exemplar der zusammengesetzten Datenbank die Datei `interfaces` mit Hilfe einer der beiden folgenden Methoden zur Verfügung:
  - Kopieren Sie diese Datei in das Verzeichnis `$HOME/sql1lib` des Exemplars der zusammengesetzten Datenbank.
  - Verwenden Sie den Befehl `ln`, um zwischen dem Verzeichnis `$sybase` und der Datei `interfaces` im Exemplarverzeichnis `$HOME/sql1lib` eine Programmverbindung herzustellen. Beispiel:

```
ln -s -f /home/sybase/interfaces /home/db2djinst1/sql1lib
```
  - Verwenden Sie die Serveroption `IFILE`, um den vollständigen Pfad für die Sybase-Datei `interfaces` anzugeben.
3. Testen Sie die Verbindung, um sicherzustellen, dass die Software von Sybase Open Client eine Verbindung zum Sybase-Server herstellen kann. Verwenden Sie ein passendes Abfragedienstprogramm, wie z. B. `isql`. Beispiel:

```
isql -Ssybnode -Umary -I/home/db2djinst1/sql1lib/interfaces
```

#### Unter Windows-Betriebssystemen:

1. Verwenden Sie zum Einrichten der Clientkonfigurationsdatei das Dienstprogramm, das im Lieferumfang der Software von Sybase Open Client enthalten ist. Diese Datei wird im Verzeichnis `%SYBASE%\ini` erstellt. Der Name der Datei lautet `sql.ini`. Weitere Informationen zur Verwendung dieses Dienstprogramms sind in der Sybase-Dokumentation enthalten.

2. Stellen Sie diese Datei `sql.ini` dem Exemplar der zusammengeschlossenen Datenbank zur Verfügung, indem Sie die Datei in das Verzeichnis `%DB2PATH%` des Exemplars der zusammengeschlossenen Datenbank kopieren. Der Standardpfad lautet `c:\Program Files\IBM\SQLLIB`.

Da DB2 Information Integrator interfaces als Standardnamen für die Sybase-Clientkonfigurationsdatei verwendet, müssen Sie die Windows-Datei `sql.ini` im Verzeichnis `c:\Program Files\IBM\SQLLIB\interfaces` umbenennen.

**Erforderlich:** Wenn Sie die Datei `sql.ini` nicht in `interfaces` umbenennen, müssen Sie beim erstellen der Serverdefinition die Serveroption `IFILE` verwenden.

3. Testen Sie die Verbindung, um sicherzustellen, dass die Software von Sybase Open Client eine Verbindung zum Sybase-Server herstellen kann. Verwenden Sie ein passendes Abfragedienstprogramm, wie z. B. **isql**. Beispiel:

```
isql -Ssybnode -Umary -I"c:\Program Files\IBM\SQLLIB\interfaces"
```

Die nächste Task in dieser Tasksequenz ist das Registrieren des Sybase-Wrappers.

#### Zugehörige Tasks:

- „Registrieren des Sybase-Wrappers“ auf Seite 392

## Registrieren des Sybase-Wrappers

Beim Hinzufügen von Sybase-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken muss der Sybase-Wrapper registriert werden.

Die Registrierung eines Wrappers ist für den Zugriff auf Sybase-Datenquellen erforderlich. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken verwendet, um mit Datenquellen zu kommunizieren und Daten aus Datenquellen abzurufen. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

#### Vorgehensweise:

Um einen Wrapper zu registrieren, setzen Sie die Anweisung `CREATE WRAPPER` ab und geben Sie den Standardnamen für den Wrapper an.

Beispiel:

```
CREATE WRAPPER CTLIB
```

**Empfehlung:** Verwenden Sie den standardmäßigen Wrappernamen `CTLIB`. Registrieren Sie einen Wrapper mit dem Standardnamen, verwendet der Server mit zusammengeschlossenen Datenbanken automatisch den standardmäßigen Bibliotheksnamen, der diesem Wrapper zugeordnet ist.

Sollte zwischen dem Namen des Wrappers und einem bereits vorhandenen Wrappernamen in der zusammengeschlossenen Datenbank ein Namenskonflikt entstehen, können Sie den Standardnamen für den Wrapper durch einen beliebigen Namen ersetzen. In diesem Fall müssen Sie in der Anweisung `CREATE WRAPPER` den Parameter `LIBRARY` angeben.

Um beispielsweise einen Wrapper mit dem Namen `sybase_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken zu registrieren, der das Betriebssystem AIX verwendet, setzen Sie die folgende Anweisung ab:

```
CREATE WRAPPER sybase_wrapper LIBRARY 'libdb2ctlib.a'
```

Der Name der von Ihnen angegebenen Wrapperbibliotheksdatei hängt von dem Betriebssystem auf dem Server mit zusammengeschlossenen Datenbanken ab. Den korrekten Namen, der in der Anweisung `CREATE WRAPPER` angegeben werden muss, finden Sie in der Liste der Sybase-Wrapper-Bibliotheksdateien.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den Sybase-Wrapper.

**Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine Sybase-Datenquelle“ auf Seite 394

**Zugehörige Referenzen:**

- „Sybase-Wrapper-Bibliotheksdateien“ auf Seite 393
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Sybase-Wrapper-Bibliotheksdateien

Die folgende Tabelle enthält eine Liste der Verzeichnispfade und Bibliotheksdateinamen für den Sybase-Wrapper.

Wenn Sie DB2 Information Integrator installieren, werden dem in der Tabelle aufgeführten Verzeichnispfad 3 Bibliotheksdateien hinzugefügt. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden die Wrapperbibliotheksdateien `libdb2ctlib.a`, `libdb2ctlibF.a` und `libdb2ctlibU.a` dem Verzeichnispfad hinzugefügt.

Wenn Sie einen Wrapper registrieren, geben Sie nur den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 83. Speicherpositionen und Dateinamen der Sybase-Wrapperbibliothek*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	<code>/usr/opt/db2_08_01/lib/</code>	<code>libdb2ctlib.a (CTLIB)</code>
HP-UX	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2ctlib.sl (CTLIB)</code>
Linux	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2ctlib.so (CTLIB)</code>
Solaris	<code>/opt/IBM/db2/V8.1/lib</code>	<code>libdb2ctlib.so (CTLIB)</code>
Windows	<code>%DB2PATH%\bin</code>	<code>db2ctlib.dll (CTLIB)</code>

`%DB2PATH%` ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows lautet `C:\Program Files\IBM\SQLLIB`.

**Zugehörige Tasks:**

- „Registrieren des Sybase-Wrappers“ auf Seite 392

## Registrieren der Serverdefinitionen für eine Sybase-Datenquelle

Beim Hinzufügen von Sybase-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen die Serverdefinitionen für eine Sybase-Datenquelle registriert werden.

In der zusammengeschlossenen Datenbank müssen Sie jeden Sybase-Server definieren, auf den Sie zugreifen möchten.

### Einschränkung:

Wenn Sie die Serverdefinitionen für eine Sybase-Datenquelle über die DB2-Steuerzentrale registrieren, muss sich die Sybase-Clientkonfigurationsdatei (*interfaces*) im jeweiligen Standardverzeichnis befinden.

- Auf Servern mit zusammengeschlossenen Datenbanken unter UNIX lautet das Standardverzeichnis `$HOME/sqllib/`.
- Auf Servern mit zusammengeschlossenen Datenbanken unter Windows lautet das Standardverzeichnis `c:\Program Files\IBM\SQLLIB`.

### Vorgehensweise:

Sie können eine Serverdefinition über die DB2-Steuerzentrale oder die DB2-Befehlszeile registrieren.

- Um diese Tasks über die DB2-Steuerzentrale auszuführen, verwenden Sie entweder den Assistenten für Objekte in zusammengeschlossenen Datenbanken oder klicken Sie den Ordner **Serverdefinitionen** mit Maustaste 2 an und klicken Sie anschließend **Erstellen** an. Das Discovery-Tool zum Aufspüren ruft die Knotennamen für die Sybase-Server ab. Sie müssen die Informationen für die Serveroption `DBNAME` angeben, um die Serverdefinition zu registrieren.
- Um diese Task über die DB2-Befehlszeile auszuführen, setzen Sie die Anweisung `CREATE SERVER` ab:

1. Suchen Sie den Knotennamen in der Sybase-Datei *interfaces*.

#### Beispiel für die Datei 'interfaces' unter UNIX-Betriebssystemen:

```
sybase119
query tcp ether anaconda 4100
```

#### Beispiel für die Datei 'interfaces' unter dem Betriebssystem Windows NT oder Windows 2000:

```
[sybase119]
query=TCP,anaconda,4100
```

In diesen Beispielen ist 'sybase119' der Knotenname. Nach dem Knotenamen steht der Verbindungstyp (TCP/IP) und der Hostname (anaconda).

Obwohl der Knotenname als Option in der Anweisung `CREATE SERVER` angegeben wird, ist er für Sybase-Datenquellen erforderlich.

2. Setzen Sie die Anweisung `CREATE SERVER` ab.

Beispiel:

```
CREATE SERVER servername TYPE SYBASE
VERSION 12.0 WRAPPER CTLIB
OPTIONS (NODE 'sybase-knoten', DBNAME 'sybase-db')
```



Verwenden Sie nach dem Erstellen der Serverdefinition die Anweisung ALTER SERVER, um Serveroptionen hinzuzufügen oder zu löschen.

**Achtung:** Wenn Sie die Datei `sql.ini` beim Einrichten der Sybase-Clientkonfigurationsdatei nicht in `interfaces` umbenannt haben, müssen Sie beim Registrieren der Serverdefinition die Serveroption `IFILE` verwenden.

Die nächste Task in dieser Tasksequenz ist das Erstellen einer Benutzerzuordnung für eine Sybase-Datenquelle.

**Zugehörige Tasks:**

- „Erstellen einer Benutzerzuordnung für eine Sybase-Datenquelle“ auf Seite 397

**Zugehörige Referenzen:**

- „ALTER SERVER statement“ in *SQL Reference, Volume 2*
- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641
- „Anweisung CREATE SERVER - Beispiele für den Sybase-Wrapper“ auf Seite 395

## Anweisung CREATE SERVER - Beispiele für den Sybase-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE SERVER verwendet wird, um Serverdefinitionen für den Sybase-Wrapper zu registrieren. Ein vollständiges Beispiel zeigt, wie eine Serverdefinition mit den erforderlichen Parametern registriert wird, und weitere Beispiele verwenden zusätzliche Serveroptionen.

**Vollständiges Beispiel:**

Das folgende Beispiel zeigt, wie eine Serverdefinition für einen Sybase-Wrapper mit Hilfe der Anweisung CREATE SERVER registriert wird:

```
CREATE SERVER SYBSERVER TYPE SYBASE VERSION 12.0 WRAPPER CTLIB
      OPTIONS (NODE 'sybase-knoten', DBNAME 'sybase-db')
```

**SYBSERVER**

Ein Name, der dem Sybase-Server zugeordnet wird. Dieser Name muss eindeutig sein. Doppelte Servernamen sind nicht zulässig.

**TYPE SYBASE**

Gibt Sybase als den Typ der Datenquelle an, für die der Zugriff konfiguriert wird. Der Parameter TYPE für den Wrapper CTLIB muss den Wert SYBASE haben.

**VERSION 12.0**

Die Version der Sybase-Datenbankserversoftware, auf die Sie zugreifen möchten. Die Versionen 11, 11.5, 11.9, 12 und 12.5 werden unterstützt.

**WRAPPER CTLIB**

Der Wrapper-Name, der in der Anweisung CREATE WRAPPER angegeben wurde.



**NODE** *'sybase-knoten'*

Der Name des Knotens, auf dem sich *SYBSERVER* befindet. Den Knotennamen finden Sie in der Datei *interfaces*. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

Obwohl der Knotenname als Option in der Anweisung *CREATE SERVER* angegeben wird, ist er für Sybase-Datenquellen erforderlich.

**DBNAME** *'sybase-db'*

Der Name der Sybase-Datenbank, auf die Sie zugreifen möchten. Diesen Namen finden Sie auf dem Sybase-Server. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

Obwohl der Datenbankname als Option in der Anweisung *CREATE SERVER* angegeben wird, ist er für Sybase-Datenquellen erforderlich.

**Achtung:** Wenn Sie die Datei *sql.ini* beim Einrichten der Sybase-Clientkonfigurationsdatei nicht in *interfaces* umbenennen, müssen Sie beim Registrieren der Serverdefinition die Serveroption *IFILE* verwenden.

**Beispiele mit zusätzlichen Serveroptionen:**

Wenn Sie den Server registrieren, können Sie in der Anweisung *CREATE SERVER* zusätzliche Serveroptionen angeben. Diese Serveroptionen umfassen allgemeine Serveroptionen sowie Serveroptionen, die für Sybase spezifisch sind.

Das folgende Beispiel zeigt, wie die Serveroption *TIMEOUT* verwendet wird, wenn Sie eine Serverdefinition auf einem Server mit zusammengeschlossenen Datenbanken registrieren, der UNIX ausführt:

```
| CREATE SERVER SYBSERVER TYPE SYBASE
|         VERSION 12.0 WRAPPER CTLIB
|         OPTIONS (NODE 'sybase-knoten', DBNAME 'sybase-db',
|                 TIMEOUT '60')
```

Der Zeitlimitwert (*TIMEOUT*) ist die Anzahl der Sekunden, die der Wrapper auf eine Antwort vom Sybase-Server wartet. Verwenden Sie die Option *TIMEOUT*, um ein gegenseitiges Sperren von Transaktionen zu vermeiden.

Das folgende Beispiel zeigt, wie die Serveroption *IFILE* verwendet wird, wenn Sie eine Serverdefinition auf einem Server mit zusammengeschlossenen Datenbanken registrieren, der Windows ausführt:

```
| CREATE SERVER SYBSERVER TYPE SYBASE
|         VERSION 12.0 WRAPPER CTLIB
|         OPTIONS (NODE 'sybase-knoten', DBNAME 'sybase-db',
|                 IFILE 'C:\Sybase\ini\sql.ini')
```

Der Wert für *IFILE* ist der vollständige Pfad und Name der Sybase Open Client-Datei *interfaces*. Verwenden Sie diese Serveroption, wenn Sie die Datei *sql.ini* nicht als *\$SQLLIB\interfaces* (auf UNIX-Systemen) oder als *%SQLLIB%/interfaces* (auf Windows-Betriebssystemen) kopiert oder verknüpft haben.

Die folgenden zusätzlichen Serveroptionen sind für Sybase spezifisch:

- *LOGIN\_TIMEOUT*
- *PACKET\_SIZE*

**Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine Sybase-Datenquelle“ auf Seite 394

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641

## Erstellen einer Benutzerzuordnung für eine Sybase-Datenquelle

Beim Hinzufügen von Sybase-Datenquellen zu Servern mit zusammenschlossenen Datenbanken muss eine Benutzerzuordnung für eine Sybase-Datenquelle erstellt werden.

Wird versucht, auf einen Sybase-Server zuzugreifen, stellt der Server mit zusammenschlossenen Datenbanken eine Verbindung zur Datenquelle her. Hierbei verwendet er eine Benutzer-ID und ein Kennwort, die für die betreffende Datenquelle gültig sind. Sie müssen eine Zuordnung (eine Benutzerzuordnung) zwischen jeder Benutzer-ID sowie jedem Kennwort des Servers mit zusammenschlossenen Datenbanken und der entsprechenden Benutzer-ID und dem entsprechenden Kennwort der Datenquelle definieren. Erstellen Sie eine Benutzerzuordnung für jede Benutzer-ID, über die auf das System zusammenschlossener Datenbanken zugegriffen wird, um verteilte Anforderungen an die Sybase-Datenquelle zu senden.

**Vorgehensweise:**

Um der Benutzer-ID und dem Kennwort des Sybase-Servers eine lokale Benutzer-ID zuzuordnen, setzen Sie die Anweisung CREATE USER MAPPING ab.

Beispiel:

```
CREATE USER MAPPING FOR benutzer-id SERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'ferne-id', REMOTE_PASSWORD 'fernes_kennwort')
```

Die nächste Task in dieser Tasksequenz ist das Testen der Verbindung zum Sybase-Server.

**Zugehörige Tasks:**

- „Testen der Verbindung zum Sybase-Server“ auf Seite 399

**Zugehörige Referenzen:**

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE USER MAPPING - Beispiele für den Sybase-Wrapper“ auf Seite 398

## Anweisung CREATE USER MAPPING - Beispiele für den Sybase-Wrapper

Dieser Abschnitt enthält Beispiele, die zeigen, wie die Anweisung CREATE USER MAPPING verwendet wird, um die Benutzer-ID eines Servers mit zusammengesetzten Datenbanken einer Benutzer-ID und einem Kennwort eines Sybase-Servers zuzuordnen. Dieser Abschnitt enthält ein vollständiges Beispiel mit den erforderlichen Parametern sowie ein Beispiel, das zeigt, wie das DB2-Sonderregister USER mit der Anweisung CREATE USER MAPPING verwendet wird.

### Vollständiges Beispiel:

Das folgende Beispiel zeigt, wie die Benutzer-ID eines Servers mit zusammengesetzten Datenbanken der Benutzer-ID und dem Kennwort eines Sybase-Servers zugeordnet wird:

```
CREATE USER MAPPING FOR mariaSERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'mary', REMOTE_PASSWORD 'day2night')
```

*maria* Gibt die lokale Benutzer-ID an, die Sie einer Benutzer-ID zuordnen, die auf dem Sybase-Server definiert wurde.

### SERVER *SYBSERVER*

Gibt den Namen des Sybase-Servers an, den Sie in der Anweisung CREATE SERVER definiert haben.

### REMOTE\_AUTHID '*mary*'

Gibt die Benutzer-ID auf dem Sybase-Server an, zu der Sie *maria* zuordnen. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption FOLD\_ID in der Anweisung CREATE SERVER nicht auf 'U' oder 'L' setzen.

### REMOTE\_PASSWORD '*day2night*'

Gibt das Kennwort an, das '*mary*' zugeordnet ist. Verwenden Sie einfache Anführungszeichen, um die Schreibweise dieses Werts beizubehalten, sofern Sie die Serveroption FOLD\_PW in der Anweisung CREATE SERVER nicht auf 'U' oder 'L' setzen.

### Beispiel für das Sonderregister:

Das folgende Beispiel zeigt eine Anweisung CREATE USER MAPPING, die das Sonderregister USER enthält:

```
CREATE USER MAPPING FOR USER SERVER SYBSERVER
    OPTIONS (REMOTE_AUTHID 'mary', REMOTE_PASSWORD 'day2night')
```

Sie können das DB2-Sonderregister USER verwenden, um die Berechtigungs-ID der Person, die die Anweisung CREATE USER MAPPING absetzt, zur Berechtigungs-ID der Datenquelle zuzuordnen, die in der Benutzeroption REMOTE\_AUTHID angegeben wurde.

### Zugehörige Tasks:

- „Erstellen einer Benutzerzuordnung für eine Sybase-Datenquelle“ auf Seite 397

### Zugehörige Referenzen:

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

## Testen der Verbindung zum Sybase-Server

Beim Hinzufügen von Sybase-Datenquellen zu Servern mit zusammengesetzten Datenbanken muss die Verbindung zum Sybase-Server getestet werden.

Sie können die Verbindung zum Sybase-Server testen, indem Sie die Serverdefinition und Benutzerzuordnungen verwenden, die Sie definiert haben.

### Vorgehensweise:

Um die Verbindung zu testen, gehen Sie wie folgt vor:

1. Öffnen Sie eine Durchgriffssitzung, um eine SQL-Anweisung SELECT für die Sybase-Systemtabellen abzusetzen.

Beispiel:

```
SET PASSTHRU name_des_fernen_servers
SELECT count(*) FROM dbo.sysobjects
SET PASSTHRU RESET
```

Hierbei ist *name\_des\_fernen\_servers* der Name, den Sie bei der Registrierung der Serverdefinition in der Anweisung CREATE SERVER für den fernen Server angegeben haben. Wenn die Anweisung SELECT eine Anzahl zurückgibt, sind Ihre Serverdefinition und Ihre Benutzerzuordnung ordnungsgemäß eingerichtet.

2. Gibt die Anweisung SELECT einen Fehler zurück, sind eventuell folgende Schritte erforderlich:
  - Den Sybase-Server prüfen, um sicherzustellen, dass er für ankommende Verbindungen konfiguriert wurde.
  - Ihre Benutzerzuordnung prüfen, um sicherzustellen, dass die Einstellungen für die Optionen REMOTE\_AUTHID und REMOTE\_PASSWORD für Verbindungen zum Sybase-Server gültig sind. Ändern Sie gegebenenfalls die Benutzerzuordnung oder erstellen Sie gegebenenfalls eine andere Benutzerzuordnung.
  - Die Sybase-Client-Software auf dem DB2-Server mit zusammengesetzten Datenbanken prüfen, um sicherzustellen, dass sie ordnungsgemäß installiert und konfiguriert wurde, um eine Verbindung zum Sybase-Server herzustellen.
  - Die Einstellungen Ihrer DB2-Variablen für zusammengesetzte Datenbanken prüfen, um sicherzustellen, dass sie für das Arbeiten mit dem Sybase-Server korrekt definiert sind. Zu diesen Variablen gehören die Systemumgebungsvariablen, die Variablen der Datei *db2dj.ini* und die Variable der DB2-Profilregistrierdatenbank (*db2set*).
  - Ihre Serverdefinition prüfen. Gegebenenfalls müssen Sie diese Definition löschen und erneut erstellen.

Die nächste Task in dieser Tasksequenz ist das Registrieren von Kurznamen für Sybase-Tabellen und -Sichten.

#### Zugehörige Tasks:

- „Hinzufügen von Sybase-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 387
- „Einrichten und Testen der Sybase-Clientkonfigurationsdatei“ auf Seite 391
- „Registrieren der Serverdefinitionen für eine Sybase-Datenquelle“ auf Seite 394
- „Registrieren von Kurznamen für Sybase-Tabellen und -Sichten“ auf Seite 400
- „Einstellen der Sybase-Umgebungsvariablen“ auf Seite 388

#### Zugehörige Referenzen:

- „ALTER USER MAPPING statement“ in *SQL Reference, Volume 2*

## Registrieren von Kurznamen für Sybase-Tabellen und -Sichten

Beim Hinzufügen von Sybase-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken müssen Kurznamen für Sybase-Tabellen und -Sichten registriert werden.

Registrieren Sie für jeden Sybase-Server, den Sie definieren, jeweils einen Kurznamen für alle Tabellen oder Sichten, auf die Sie zugreifen wollen. Verwenden Sie diese Kurznamen anstelle der Namen der Datenquellenobjekte, wenn Sie die Sybase-Server abfragen.

Die zusammengeschlossene Datenbank verwendet Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Diese Statistiken werden gesammelt, wenn Sie unter Verwendung der Anweisung CREATE NICKNAME einen Kurznamen für ein Datenquellenobjekt registrieren. Die zusammengeschlossene Datenbank stellt das Vorhandensein des Objekts in der Datenquelle sicher und versucht dann, bestehende statistische Daten zur Datenquelle zu sammeln. Informationen, die für das Optimierungsprogramm nützlich sind, werden aus den Katalogen der Datenquelle gelesen und in den globalen Katalog auf dem Server mit zusammengeschlossenen Datenbanken aufgenommen. Da einige oder alle der Kataloginformationen von Datenquellen vom Optimierungsprogramm verwendet werden können, sollten Sie die Statistiken (unter Verwendung des Datenquellenbefehls, der dem DB2-Befehl RUNSTATS entspricht) in der Datenquelle aktualisieren, bevor Sie einen Kurznamen registrieren.

#### Vorgehensweise:

Um einen Kurznamen zu registrieren, setzen Sie die Anweisung CREATE NICKNAME ab.

Beispiel:

```
CREATE NICKNAME sybase-name FOR SYBSERVER."remote_schema".remote.table"
```

Kurznamen können bis zu 128 Zeichen lang sein.

Wiederholen Sie diesen Schritt für jede Sybase-Tabelle oder -Sicht, für die Sie einen Kurznamen erstellen möchten.

Wenn Sie den Kurznamen erstellen, verwendet der Server mit zusammengeschlossenen Datenbanken die von Ihnen angegebenen Informationen, um den Datenquellenkatalog abzufragen. Diese Abfrage testet Ihre Verbindung zur Datenquelle unter Verwendung des Kurznamens. Wenn die Verbindung nicht funktioniert, empfangen Sie eine Fehlermeldung.

**Zugehörige Referenzen:**

- „RUNSTATS Command“ in *Command Reference*
- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den Sybase-Wrapper“ auf Seite 401

## **Anweisung CREATE NICKNAME - Beispiele für den Sybase-Wrapper**

Dieser Abschnitt enthält ein Beispiel, das zeigt, wie die Anweisung CREATE NICKNAME verwendet wird, um einen Kurznamen für eine Sybase-Tabelle oder -Sicht zu registrieren, auf die Sie zugreifen möchten.

Dieses Beispiel zeigt, wie ein fernes Objekt für den Sybase-Server angegeben wird, unter dem der Kurzname zugeordnet ist:

```
CREATE NICKNAME SYBSALES FOR SYBSERVER."salesdata"."europe"
```

**SYBSALES**

Eindeutiger Kurzname für die Sybase-Tabelle oder -Sicht.

Der Kurzname besteht aus zwei Teilen - dem Schema und dem Kurznamen. Wenn Sie bei der Erstellung des Kurznamens das Schema auslassen, wird die Berechtigungs-ID des Benutzers, der den Kurznamen erstellt, zum Schema des Kurznamens.

**SYBSERVER."salesdata"."europe"**

Eine dreiteilige Kennung für das ferne Objekt.

- SYBSERVER ist der Name, den Sie dem Sybase-Datenbankserver in der Anweisung CREATE SERVER zugeordnet haben.
- salesdata ist der Name des fernen Schemas, zu dem die Tabelle oder Sicht gehört.
- europe ist der Name der fernen Tabelle oder Sicht, auf die Sie zugreifen möchten.

Der Server mit zusammengeschlossenen Datenbanken setzt die Namen der Sybase-Schemata und -Tabellen in Großbuchstaben um, sofern Sie die Namen nicht in Anführungszeichen setzen.

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Sybase-Tabellen und -Sichten“ auf Seite 400

**Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

---

## Optimierung und Fehlerbehebung Konfiguration für Sybase-Datenquellen

Nachdem Sie die Konfiguration für Sybase-Datenquellen eingerichtet haben, müssen Sie diese Konfiguration möglicherweise ändern, um die Leistung zu verbessern.

### Auflösen des Fehlers `sp_helpindex`

Das System zusammengesetzter Datenbanken basiert auf einer der gespeicherten Prozeduren des Sybase-Katalogs: `sp_helpindex`. Empfangen Sie den folgenden SQL-Fehler, sind die gespeicherten Prozeduren des Sybase-Katalogs möglicherweise nicht auf dem Sybase-Server installiert.

```
SQL0204N "sp_helpindex" is an undefined name.
```

In diesem Fall muss die Installation der gespeicherten Prozeduren des Katalogs auf dem Sybase-Server durch den Sybase-Administrator ausgeführt werden.

#### Zugehörige Tasks:

- „Hinzufügen von Sybase-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 387

#### Zugehörige Referenzen:

- „db2set - DB2 Profile Registry Command“ in *Command Reference*

---

## Kapitel 21. Konfigurieren des Zugriffs auf Datenquellen in Dateien mit Tabellenstruktur

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Datenquellen in Dateien mit Tabellenstruktur gespeichert sind. Der Zugriff auf Datenquellen in Dateien mit Tabellenstruktur kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält folgende Informationen:

- Erläuterungen zu Dateien mit Tabellenstruktur.
- Liste der auszuführenden Tasks.
- Beispiele zu den benötigten SQL-Anweisungen.
- Liste der dem Wrapper für Dateien mit Tabellenstruktur zugeordneten Fehlermeldungen.

---

### Was sind Dateien mit Tabellenstruktur?

Eine Datei mit Tabellenstruktur besitzt eine regelmäßige Struktur, die aus einer Reihe von Datensätzen besteht, wobei jeder Datensatz die gleiche Anzahl von Feldern enthält, die jeweils durch ein willkürlich gewähltes Begrenzungszeichen getrennt sind. Nullwerte werden durch zwei unmittelbar benachbarte Begrenzungszeichen dargestellt.

Das folgende Beispiel zeigt den Inhalt einer Datei mit dem Namen DRUGDATA1.TXT. Sie enthält drei Datensätze, jeder mit drei Feldern, die durch Kommas voneinander getrennt sind:

```
234,DrugnameA,Manufacturer1  
332,DrugnameB,Manufacturer2  
333,DrugnameC,Manufacturer2
```

Das erste Feld ist die eindeutige ID-Nummer für das jeweilige Medikament (Drug). Das zweite Feld enthält den Namen des Medikaments. Das dritte Feld enthält den Namen des Herstellers, der das Medikament produziert.

#### Zugehörige Konzepte:

- „Attribute von Dateien mit Tabellenstruktur“ auf Seite 404
- „Arbeiten mit Dateien mit Tabellenstruktur in DB2 Information Integrator“ auf Seite 404

#### Zugehörige Tasks:

- „Hinzufügen von Datenquellen von Dateien mit Tabellenstruktur zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 406



---

## Attribute von Dateien mit Tabellenstruktur

Dateien mit Tabellenstruktur können sortiert oder unsortiert sein. Der Wrapper von Dateien mit Tabellenstruktur kann sortierte Datendateien wesentlich effizienter durchsuchen als unsortierte Dateien.

### Sortierte Dateien

Die Datei DRUGDATA1.TXT enthält sortierte Datensätze. Die Datei ist nach dem ersten Feld, das heißt, der eindeutigen ID-Nummer des Medikaments sortiert. Dieses Feld ist der Primärschlüssel, weil es für jedes Medikament eindeutig ist. Sortierte Dateien müssen in aufsteigender Reihenfolge sortiert sein.

```
234,DrugnameA,Manufacturer1
332,DrugnameB,Manufacturer2
333,DrugnameC,Manufacturer2
```

### Unsortierte Dateien

Die Datei DRUGDATA2.TXT enthält unsortierte Datensätze. Für die Reihenfolge, in der die Datensätze in der Datei aufgelistet sind, gilt keine bestimmte Ordnung.

```
332,DrugnameB,Manufacturer2
234,DrugnameA,Manufacturer1
333,DrugnameC,Manufacturer2
```

#### Zugehörige Konzepte:

- „Was sind Dateien mit Tabellenstruktur?“ auf Seite 403
- „Arbeiten mit Dateien mit Tabellenstruktur in DB2 Information Integrator“ auf Seite 404

#### Zugehörige Tasks:

- „Hinzufügen von Datenquellen von Dateien mit Tabellenstruktur zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 406

---

## Arbeiten mit Dateien mit Tabellenstruktur in DB2 Information Integrator

Mit Hilfe eines als Wrapper bezeichneten Moduls kann DB2<sup>®</sup> Information Integrator SQL-Anweisungen, die Daten in einer Datei mit Tabellenstruktur abfragen, so verarbeiten, als wären die Daten in einer normalen relationalen Tabelle oder Sicht enthalten. Dadurch können Daten in einer Datei mit Tabellenstruktur mit relationalen Daten oder Daten in anderen Dateien mit Tabellenstruktur verknüpft werden. Dieser Prozess wird in Abb. 27 auf Seite 405 dargestellt.

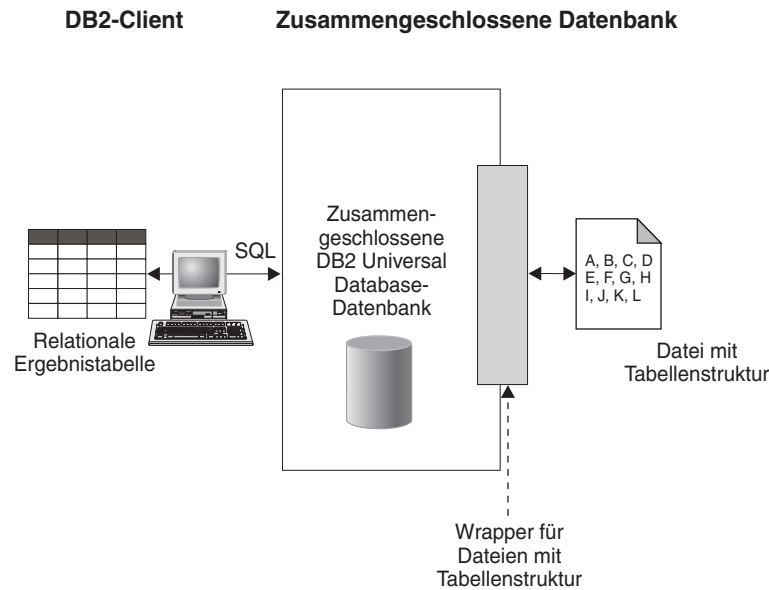


Abbildung 27. Funktionsweise des Wrappers für Dateien mit Tabellenstruktur

Nehmen Sie zum Beispiel an, dass die Datei DRUGDATA2.TXT, die eine Tabellenstruktur aufweist, auf Ihrem Computer im Labor gespeichert ist. Ein Versuch, diese Daten abzufragen und mit anderen, von Ihnen verwendeten Tabellen aus anderen Datenquellen abzugleichen, kann sehr mühsam sein.

Nachdem Sie die Datei DRUGDATA2.TXT bei DB2 Information Integrator registriert haben, verhält sich die Datei wie eine relationale Datenquelle. Sie können die Datei nun zusammen mit anderen relationalen und nicht relationalen Datenquellen abfragen und die Dateien im Verbund analysieren.

Sie könnten zum Beispiel die folgende Abfrage ausführen:

```
SELECT * FROM DRUGDATA2 ORDER BY DCODE
```

Diese Abfrage liefert die folgenden Ergebnisse.

Dcode	Drug	Manufacturer
234	DrugnameA	Manufacturer1
332	DrugnameB	Manufacturer2
333	DrugnameC	Manufacturer2

#### Zugehörige Konzepte:

- „Was sind Dateien mit Tabellenstruktur?“ auf Seite 403
- „Attribute von Dateien mit Tabellenstruktur“ auf Seite 404

#### Zugehörige Tasks:

- „Hinzufügen von Datenquellen von Dateien mit Tabellenstruktur zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 406

---

## Hinzufügen von Dateien mit Tabellenstruktur zu einem System zusammengesetzter Datenbanken

### Hinzufügen von Datenquellen von Dateien mit Tabellenstruktur zu einem Server mit zusammengesetzten Datenbanken

Um den Server mit zusammengesetzten Datenbanken für den Zugriff auf Datenquellen von Dateien mit Tabellenstruktur zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengesetzten Datenbanken auf Datenquellen von Dateien mit Tabellenstruktur mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengesetzten Datenbanken führt.

#### Voraussetzungen:

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengesetzten Datenbanken fungiert.
- Eine zusammengesetzte Datenbank muss sich auf dem Server mit zusammengesetzten Datenbanken befinden.

#### Vorgehensweise:

Führen Sie folgende Schritte aus, um Datenquellen von Dateien mit Tabellenstruktur zum Server mit zusammengesetzten Datenbanken hinzuzufügen:

1. Registrieren Sie den Wrapper.
2. Registrieren Sie die Serverdefinition.
3. Registrieren Sie die Kurznamen für die Dateien mit Tabellenstruktur.

#### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

#### Zugehörige Tasks:

- „Erstellen einer zusammengesetzten Datenbank“ auf Seite 58
- „Registrieren des Wrappers für Dateien mit Tabellenstruktur“ auf Seite 407
- „Registrieren des Servers für Dateien mit Tabellenstruktur“ auf Seite 408
- „Registrieren der Kurznamen für Dateien mit Tabellenstruktur“ auf Seite 408

#### Zugehörige Referenzen:

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## Registrieren des Wrappers für Dateien mit Tabellenstruktur

Die Registrierung des Wrappers für Dateien mit Tabellenstruktur ist ein Teil der umfangreicheren Task des Hinzufügens von Datenquellen von Dateien mit Tabellenstruktur zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf Datenquellen von Dateien mit Tabellenstruktur registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

### Vorgehensweise:

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Bibliotheksdatei des Wrappers aus.

Um beispielsweise einen Wrapper mit dem Namen `laboratory_flat_files` auf einem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER laboratory_flat_files LIBRARY 'libdb2lsfile.a';
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung CREATE WRAPPER finden Sie in der Liste der Bibliotheksdateien von Wrappern für Dateien mit Tabellenstruktur.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinition für die Dateien mit Tabellenstruktur.

### Zugehörige Referenzen:

- „Bibliotheksdateien des Wrappers für Dateien mit Tabellenstruktur“ auf Seite 407
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Bibliotheksdateien des Wrappers für Dateien mit Tabellenstruktur

Die folgende Tabelle enthält eine Liste der Verzeichnispfade und Bibliotheksdateinamen für den Wrapper für Dateien mit Tabellenstruktur.

Wenn Sie DB2 Information Integrator installieren, werden dem in der Tabelle aufgeführten Verzeichnispfad 3 Bibliotheksdateien hinzugefügt. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden die Wrapperbibliotheksdateien `libdb2lsfile.a`, `libdb2lsfileF.a` und `libdb2lsfileU.a` dem Verzeichnispfad hinzugefügt.

Wenn Sie einen Wrapper registrieren, geben Sie nur den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

Tabelle 84. Speicherpositionen und Dateinamen der Bibliothek des Wrappers für Dateien mit Tabellenstruktur

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2lsfile.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2lsfile.sl
Linux	/opt/IBM/db2/V8.1/lib	libdb2lsfile.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lsfile.so
Windows	%DB2PATH%\bin	db2lsfile.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows lautet C:\Program Files\IBM\SQLLIB.

**Zugehörige Tasks:**

- „Registrieren des Wrappers für Dateien mit Tabellenstruktur“ auf Seite 407

## Registrieren des Servers für Dateien mit Tabellenstruktur

Die Registrierung des Servers für Dateien mit Tabellenstruktur ist der Teil der umfangreicheren Task des Hinzufügens von Dateien mit Tabellenstruktur zu einem System zusammengesetzter Datenbanken. Nach der Registrierung des Wrappers muss ein entsprechender Server registriert werden.

**Vorgehensweise:**

Um den Server für Dateien mit Tabellenstruktur für das System zusammengesetzter Datenbanken zu registrieren, verwenden Sie die Anweisung CREATE SERVER. Zum Beispiel:

```
CREATE SERVER biochem_lab WRAPPER laboratory_flat_files
```

In diesem Beispiel ist biochem\_lab der Name, der dem Server für die Dateien mit Tabellenstruktur zugeordnet ist. Der Name muss innerhalb der Datenbank, für die der Server registriert wird, eindeutig sein.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für für Dateien mit Tabellenstruktur.

**Zugehörige Tasks:**

- „Registrieren der Kurznamen für Dateien mit Tabellenstruktur“ auf Seite 408

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

## Registrieren der Kurznamen für Dateien mit Tabellenstruktur

Die Registrierung der Kurznamen für Dateien mit Tabellenstruktur ist der Teil der umfangreicheren Task des Hinzufügens von Dateien mit Tabellenstruktur zu einem System zusammengesetzter Datenbanken. Nach der Registrierung eines Servers muss ein entsprechender Kurzname registriert werden. Kurznamen werden in Abfragen zum Verweisen auf Datenquellen mit Dateien mit Tabellenstruktur verwendet.

Für die Zuordnung von Kurznamen zu Dateien mit Tabellenstruktur gibt es zwei Methoden:

- Die fixierte Methode unter Verwendung der Kurznamenoption FILE\_PATH. Wird diese Option verwendet, stellt der Kurzname Daten aus einer spezifischen Datei mit Tabellenstruktur dar.
- Die Methode mit einem bei der Abfrage angegebenen Dateinamen unter Verwendung der Kurznamenspaltenoption DOCUMENT. Wird diese Option verwendet, kann der Kurzname zur Darstellung von Daten aus beliebigen Dateien mit Tabellenstruktur verwendet werden, deren Schema der Kurznamendefinition entspricht.

#### **Einschränkungen:**

Wenn ein nichtnumerisches Feld für seinen Spaltentyp zu lang ist, werden die überzähligen Zeichen abgeschnitten. Wenn ein Dezimalfeld in der Datei mehr Stellen nach dem Basiszeichen besitzt, als durch den Parameter für die Kommastellenzahl des Spaltentyps zugelassen werden, werden die überzähligen Zeichen abgeschnitten. Das Basiszeichen (Radixzeichen) wird durch das RADIXCHAR-Element der NLS-Kategorie LC\_NUMERIC festgelegt (NLS - National Language Support).

Die maximale Zeilenlänge beträgt 10 MB (10485760 Byte).

#### **Vorgehensweise:**

Um einen Kurznamen zu registrieren, verwenden Sie die Anweisung CREATE NICKNAME für jede Datei mit Tabellenstruktur, auf die Sie zugreifen wollen.

Dies war die letzte Task in dieser Tasksequenz.

#### **Zugehörige Tasks:**

- „Hinzufügen von Datenquellen von Dateien mit Tabellenstruktur zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 406
- „Angaben von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75

#### **Zugehörige Referenzen:**

- „Modell der Dateizugriffssteuerung für den Wrapper für Dateien mit Tabellenstruktur“ auf Seite 410
- „Tipps und Überlegungen für die Optimierung des Wrappers für Dateien mit Tabellenstruktur“ auf Seite 411
- „Syntax der Anweisung CREATE NICKNAME - Wrapper für Dateien mit Tabellenstruktur“ auf Seite 618
- „Anweisung CREATE NICKNAME - Beispiele für den Wrapper für Dateien mit Tabellenstruktur“ auf Seite 409

## **Anweisung CREATE NICKNAME - Beispiele für den Wrapper für Dateien mit Tabellenstruktur**

Dieser Abschnitt enthält ein vollständiges Beispiel, das zeigt, wie die Anweisung CREATE NICKNAME verwendet wird, um Kurznamen für den Wrapper für Dateien mit Tabellenstruktur zu registrieren. Außerdem enthält der Abschnitt Beispiele für spezifische Optionen.

### Vollständiges Beispiel:

Das folgende Beispiel zeigt eine Anweisung CREATE NICKNAME für die Datei mit Tabellenstruktur DRUGDATA1.TXT:

```
CREATE NICKNAME DRUGDATA1(Dcode Integer NOT NULL, Drug CHAR(20),
  Manufacturer CHAR(20))
FOR SERVER biochem_lab OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT',
  COLUMN_DELIMITER ',', SORTED 'Y', KEY_COLUMN 'DCODE', VALIDATE_DATA_FILE 'Y')
```

### Beispiele für die Option KEY COLUMN:

Dieses Beispiel zeigt, dass die Spalte so ausgewiesen ist, dass sie keine Nullwerte enthalten darf, indem der Definition der Spalte in der Kurznamenanweisung die Option NOT NULL hinzugefügt wird:

```
CREATE NICKNAME tox (tox_id INTEGER NOT NULL, toxicity VARCHAR(100))
FOR SERVER tox_server1
  OPTIONS (FILE_PATH'/tox_data.txt', SORTED 'Y')
```

```
CREATE NICKNAME weights (mol_id INTEGER, wt VARCHAR(100) NOT NULL)
FOR SERVER wt_server
  OPTIONS (FILE_PATH'/wt_data.txt', SORTED 'Y', KEY_COLUMN 'WT')
```

Diese Option ist abhängig von der Groß-/Kleinschreibung. Allerdings setzt DB2 die Spaltennamen in Großbuchstaben um, sofern die Spalte nicht mit doppelten Anführungszeichen definiert wird. Das folgende Beispiel funktioniert nicht einwandfrei, weil der Name der Spalte empno von DB2 in Großbuchstaben umgewandelt wird und die Spalte empno in Kleinbuchstaben übergeben wird. Daher wird die als Schlüssel ausgewiesene Spalte nicht gefunden.

```
CREATE NICKNAME depart (
  empno char(6) NOT NULL)
FOR SERVER DATASTORE
  OPTIONS(FILE_PATH'data.txt', SORTED 'Y', KEY_COLUMN 'empno');
```

### Zugehörige Tasks:

- „Registrieren der Kurznamen für Dateien mit Tabellenstruktur“ auf Seite 408

### Zugehörige Referenzen:

- „Syntax der Anweisung CREATE NICKNAME - Wrapper für Dateien mit Tabellenstruktur“ auf Seite 618

---

## Modell der Dateizugriffssteuerung für den Wrapper für Dateien mit Tabellenstruktur

Das Datenbankverwaltungssystem greift auf Dateien mit Tabellenstruktur mit der Berechtigung des DB2-Exemplareigners zu. Der Wrapper kann nur auf Dateien zugreifen, die von dieser Benutzer-ID (oder Gruppen-ID) gelesen werden können. Die Berechtigungs-ID der Anwendung (d. h. die ID, die die Verbindung zur zusammengeschlossenen Datenbank herstellt) ist nicht relevant.

Bei DB2 Universal Database Enterprise Server Edition muss es möglich sein, auf jede Datei mit Tabellenstruktur, für die ein Kurzname erstellt wurde, von jedem Knoten aus über denselben Pfad zuzugreifen. Die Datei muss sich nicht auf einem DB2 Universal Database-Knoten befinden, vorausgesetzt, es kann von jedem beliebigen Knoten über einen gemeinsamen Pfad auf sie zugegriffen werden.

Zum Zugriff in einem zugeordneten Laufwerk auf eine Datei mit Tabellenstruktur bei einer konfigurierten Windows NT- oder Windows 2000-Domäne im Netz muss das DB2-Serviceanmeldekonto sich in der Domäne befinden, die über Zugriff auf den gemeinsam benutzten Ordner mit den Datendateien verfügt.

Zum Zugriff in einem zugeordneten Laufwerk auf eine Datei mit Tabellenstruktur bei nicht konfiguierter Windows NT- oder Windows 2000-Domäne im Netz und lokaler Benutzeranmeldung an jeder Workstation sollte das DB2-Serviceanmeldekonto denselben Benutzernamen und dasselbe Kennwort haben wie ein gültiger Benutzer der Maschine, die den Ordner gemeinsam benutzt. Dieser Benutzer muss auf der Berechtigungsliste für den gemeinsam benutzten Ordner mindestens mit Lesezugriff enthalten sein.

**Zugehörige Referenzen:**

- „Zugriffssteuerung für den Documentum-Wrapper“ auf Seite 214
- „Modell für die Dateizugriffssteuerung für den Excel-Wrapper“ auf Seite 256

## Tipps und Überlegungen für die Optimierung des Wrappers für Dateien mit Tabellenstruktur

- Das System kann sortierte Datendateien wesentlich effizienter als unsortierte Dateien durchsuchen.
- Für sortierte Dateien können Sie den Durchsatz erhöhen, indem Sie beim Übergeben einer Abfrage einen Wert oder einen Wertebereich für die Spaltenspalte angeben.
- Die Statistikdaten für Kurznamen von Dateien mit Tabellenstruktur müssen manuell in den SYSSTAT- und SYSCAT-Sichten aktualisiert werden.

**Zugehörige Referenzen:**

- „Tipps für die Optimierung des BLAST-Wrappers“ auf Seite 134

## Nachrichten für den Wrapper für Dateien mit Tabellenstruktur

Dieser Abschnitt enthält eine Liste von Nachrichten und zugehörigen Beschreibungen, die Ihnen bei der Arbeit mit dem Wrapper für Dateien mit Tabellenstruktur eventuell begegnen können.

*Tabelle 85. Vom Wrapper für Dateien mit Tabellenstruktur generierte Nachrichten*

Fehlercode	Nachricht	Erläuterung
SQL0405N	Das numerische Literal "<literal>" ist nicht gültig, da sein Wert außerhalb des zulässigen Bereichs liegt.	Eine Spalte in der Datendatei oder ein Vergleichselementwert in einer SQL-Anweisung enthält einen Wert, der außerhalb des zulässigen Wertebereichs für den Datentyp liegt. Berichten Sie die Datendatei, oder definieren Sie die Spalte in einen geeigneteren Datentyp um.
SQL0408N	Ein Wert ist mit dem Datentyp seines Zuordnungsziels nicht kompatibel. Der Name des Ziels ist "<spaltenname>".	Eine Spalte in der Datendatei enthält Zeichen, die für den verwendeten Datentyp ungültig sind. Berichten Sie die Datendatei, oder definieren Sie die Spalte in einen geeigneteren Datentyp um.



Tabelle 85. Vom Wrapper für Dateien mit Tabellenstruktur generierte Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Datenquellenpfad ist NULL.")	Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Abrufen der Schlüsselspalte fehlgeschlagen.")	Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "STAT für Datenquelle fehlgeschlagen. ERRNO = <fehlernummer>.")	Stellen Sie sicher, dass Sie über die entsprechenden Verzeichnisberechtigungen verfügen. Stellen Sie sicher, dass die Datei vorhanden ist.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Keine Spalteninformationen gefunden.")	Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Nicht unterstützter Operator.")	Wenden Sie sich an die IBM Softwareunterstützung.
SQL1816N	Der Wrapper "<wrappername>" kann nicht für den Zugriff auf den "Typ" von Datenquelle ("<typ>") verwendet werden, den Sie für die zusammengeslossene Datenbank zu definieren versuchen.	Der Servertyp war ungültig. In der Anweisung CREATE SERVER darf kein Servertyp angegeben werden. Entfernen Sie das Schlüsselwort und den Wert für TYPE und wiederholen Sie die Anweisung.

Tabelle 85. Vom Wrapper für Dateien mit Tabellenstruktur generierte Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "ERRNO = <fehlernummer>" von der Datenquelle "<servername>" empfangen. Zugeordneter Text und Token sind "Datei kann nicht gelesen werden".	Überprüfen Sie den Wert der Fehlernummer. Stellen Sie sicher, dass die Datei durch den DB2-Exemplareigner gelesen werden kann. Führen Sie anschließend den SQL-Befehl erneut aus.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Datenfehler" von der Datenquelle "<servername>" empfangen. Zugeordneter Text und Token sind "Datenquelle ist eine vom Standard abweichende Datei".	Die Datenquellendatei ist ein Verzeichnis, ein Socket oder ein FIFO-Stapel. Der Zugriff ist nur auf Standarddateien als Datenquellen möglich. Ändern Sie die Dateipfadoption FILE_PATH, sodass sie auf eine gültige Datei zeigt, und führen Sie den SQL-Befehl erneut aus.
SQL1822N	Es wurde ein unerwarteter Fehlercode "ERRNO = <fehlernummer>" von der Datenquelle "<servername>" empfangen. Zugeordneter Text und Token sind "Fehler beim Öffnen der Datei".	Der Wrapper konnte die Datei nicht öffnen. Überprüfen Sie die Fehlernummer, um die Ursache des Fehlers festzustellen. Beheben Sie das Problem mit der Datenquelle, und führen Sie den SQL-Befehl erneut aus.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Datenfehler" von der Datenquelle "<servername>" empfangen. Zugeordneter Text und Token sind "Schlüsselspalte fehlt".	In einem aus der Datenquelle abgerufenen Datensatz fehlte der Wert für das Schlüsselfeld. Die Schlüsselspalte darf keinen Nullwert enthalten. Korrigieren Sie die Daten oder registrieren Sie die Datei mit einem unsortierten Kurznamen.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Datenfehler" von der Datenquelle "<servername>" empfangen. Zugeordneter Text und Token sind "Datei nicht sortiert".	Die Datei war nicht nach der Schlüsselspalte sortiert. Führen Sie eine der folgenden Maßnahmen durch: Ändern Sie die Option KEY_COLUMN, damit sie auf die richtige Spalte verweist, sortieren Sie die Datendatei erneut, oder registrieren Sie den Kurznamen als unsortierten Kurznamen.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Datenfehler" von der Datenquelle "<servername>" empfangen. Zugeordneter Text und Token sind "Schlüssel überschreitet Definitionsgröße".	Das aus der Datenquelle gelesene Feld der Schlüsselspalte war größer als die DB2-Spaltendefinition, was dazu führen kann, dass die Suchroutinen des Wrappers nicht korrekt funktionieren. Berichten Sie die Daten oder die Kurznamendefinition, und registrieren Sie den Kurznamen erneut.
SQL1822N	Es wurde ein unerwarteter Fehlercode "Datenfehler" von der Datenquelle "<servername>" empfangen. Zugeordneter Text und Token sind "Zeile in Datendatei länger als 32 KB".	Eine Zeile in der Datendatei überschreitet die maximale Zeilenlänge, die vom Wrapper zugelassen wird. Die Zeilenlänge darf 32768 Byte nicht überschreiten. Kürzen Sie die Zeile in der Datendatei.
SQL1823N	Für den Datentyp "<datentyp>" gibt es vom Server "<servername>" keine Datentypzuordnung.	Der Kurzname wurde mit einem nicht unterstützten Datentyp definiert. Definieren Sie den Kurznamen unter Verwendung nur unterstützter Datentypen erneut.

Tabelle 85. Vom Wrapper für Dateien mit Tabellenstruktur generierte Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1881N	"<optionsname>" ist keine gültige Option der "<komponente>" für "<objektname>".	Der aufgeführte Wert ist keine gültige Option für das aufgeführte Objekt. Entfernen oder ändern Sie die ungültige Option, und übergeben Sie die SQL-Anweisung anschließend erneut.
SQL1882N	Die Option "COLUMN_DELIMITER" der Art "NICKNAME" kann nicht auf "<spaltenbegrenzer>" für "<name_des_kurznamens>" eingestellt werden.	Der Spaltenbegrenzer war länger als ein Zeichen. Definieren Sie die Option mit nur einem Zeichen erneut. Führen Sie anschließend die SQL-Anweisung erneut aus.
SQL1882N	Die Option "KEY_COLUMN" der Art "NICKNAME" kann nicht auf "<spaltenname>" für "<name_des_kurznamens>" eingestellt werden.	Die als Schlüsselspalte ausgewählte Spalte ist für diesen Kurznamen nicht definiert. Berichtigen Sie die Option KEY_COLUMN, sodass sie eine der sortierten Spalten für den Kurznamen ist, und führen Sie den SQL-Befehl erneut aus.
SQL1882N	Die Option "VALIDATE_DATA_FILE" der Art "NICKNAME" kann nicht auf "<optionswert>" für "<name_des_kurznamens>" eingestellt werden.	Der angegebene Wert der Option war ungültig. Gültige Werte sind "Y" bzw. "N". Berichtigen Sie die Option, und registrieren Sie den Kurznamen erneut.
SQL1883N	"<optionsname>" ist eine erforderliche Option der Art "<komponente>" für "<objektname>".	In der SQL-Anweisung fehlte eine erforderliche Option für den Wrapper. Fügen Sie die erforderliche Option hinzu, und übergeben Sie die SQL-Anweisung erneut.
SQL30090N	Die Operation ist für die Umgebung der Anwendungsausführung nicht gültig. Ursachencode = "21".	Sie haben eine Durchgriffssitzung versucht. Der Wrapper für Dateien mit Tabellenstruktur unterstützt keine Durchgriffssitzungen.

**Zugehörige Konzepte:**

- „Nachrichten - Einführung“ in *Fehlernachrichten Band 1*

**Zugehörige Referenzen:**

- „SQLSTATE-Nachrichten“ in *Fehlernachrichten Band 2*

---

## Kapitel 22. Konfigurieren des Zugriffs auf Teradata-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in Teradata-Datenquellen gespeichert sind. Der Zugriff auf Teradata-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält eine Liste der Tasks, die Sie ausführen müssen, wenn Sie den Server mit zusammengeschlossenen Datenbanken konfigurieren, zeigt Beispiele zu den benötigten SQL-Anweisungen und bietet Informationen zur Optimierung und Fehlerbehebung beim Konfigurieren des Servers mit zusammengeschlossenen Datenbanken.

---

### Hinzufügen von Teradata zu einem System zusammengeschlossener Datenbanken

#### Hinzufügen von Teradata-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Teradata-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengeschlossenen Datenbanken auf Teradata-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

##### Voraussetzungen:

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.
- Die Teradata-Client-Software muss auf dem Server mit zusammengeschlossenen Datenbanken installiert und konfiguriert sein.
  - Um auf Teradata Version V2R5 zuzugreifen, muss der Teradata-Client die Teradata-CLI Version 2 (CLIV2), Release 04.07 (oder höher) unterstützen.
  - Um auf Teradata Version V2R3 oder V2R4 zuzugreifen, muss der Teradata-Client die Teradata-CLI Version 2 (CLIV2), Release 04.06 (oder höher) unterstützen.

##### Vorgehensweise:

Führen Sie folgende Schritte aus, um Teradata-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen:

1. Optional: Testen Sie die Verbindung zum Teradata-Server.

2. Überprüfen Sie, ob die Teradata-Bibliothek für das Binden zur Laufzeit aktiviert ist (AIX).
3. Setzen Sie die Umgebungsvariablen für den Teradata-Wrapper.
4. Registrieren Sie den Wrapper.
5. Registrieren Sie die Serverdefinition.
6. Erstellen Sie die Benutzerzuordnungen.
7. Testen Sie die Verbindung zwischen dem Server mit zusammengeschlossenen Datenbanken und dem Teradata-Server.
8. Registrieren Sie Kurznamen für die Teradata-Tabellen und -Sichten.

#### **Zugehörige Konzepte:**

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Konfigurieren von Datenquellen - Kurzüberblick“ auf Seite 63

#### **Zugehörige Tasks:**

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Optimierung und Fehlerbehebung der Konfiguration für Teradata-Datenquellen“ auf Seite 430
- „Testen der Verbindung zum Teradata-Server“ auf Seite 416
- „Registrieren des Teradata-Wrappers“ auf Seite 421
- „Registrieren der Serverdefinitionen für eine Teradata-Datenquelle“ auf Seite 422
- „Erstellen der Benutzerzuordnung für eine Teradata-Datenquelle“ auf Seite 425
- „Registrieren von Kurznamen für Teradata-Tabellen und -Sichten“ auf Seite 428
- „Überprüfung der Konfiguration des Servers mit zusammengeschlossenen Datenbanken“ auf Seite 43
- „Einstellen der Teradata-Umgebungsvariablen“ auf Seite 418

#### **Zugehörige Referenzen:**

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## **Testen der Verbindung zum Teradata-Server**

Das Testen der Verbindung zum Teradata-Server ist Teil der umfangreicheren Task des Hinzufügens von Teradata-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken.

Bevor Sie einen Wrapper, eine Serverdefinition oder eine Benutzerzuordnung erstellen, können Sie die Verbindung zum Teradata-Server testen. Überprüfen Sie zuerst die Verbindung, um sicherzustellen, dass die Clientsoftware korrekt eingerichtet ist, und um Fehler beim Ausgeben der Anweisungen CREATE WRAPPER, CREATE SERVER und CREATE USER MAPPING zu vermeiden.

Mit dem Dienstprogramm Basic Teradata Query (BTEQ) können Sie eine SQL-Abfrage übergeben, um die Verbindung zum Teradata-Server zu überprüfen. Weitere Informationen über das Dienstprogramm BTEQ sind in der Teradata-Dokumentation enthalten.

### Voraussetzungen:

Stellen Sie sicher, dass das Dienstprogramm BTEQ und das Teradata Data Connector Application Program Interface (PIOM) während des Installationsprozesses der Teradata- Clientsoftware installiert wurden.

### Vorgehensweise:

Gehen Sie wie folgt vor, um die Verbindung zum Teradata-Server zu testen:

1. Starten Sie eine Sitzung des Dienstprogramms BTEQ, und melden Sie sich am Teradata- Server an.
2. Geben Sie einen SQL-Befehl aus, um sicherzustellen, dass Sie eine Verbindung zum Teradata-Server herstellen können. Beispiel:

```
select count(*) from dbc.tables;
```

Wurde die Verbindung erfolgreich hergestellt, sollten die Ausgabe der Abfrage auf der Anzeige zu sehen sein. Beispiel:

```
*** Query completed. One row found. One column returned.  
*** Total elapsed time was 1 second.
```

```
Count(*)
```

```
-----  
497
```

Wurde die Verbindung nicht erfolgreich hergestellt, überprüfen Sie die Teradata- Clientsoftware, um sicherzustellen, dass diese auf dem Server mit zusammengeschlossenen Datenbanken korrekt installiert und konfiguriert wurde.

3. Melden Sie sich vom Teradata-Server ab, und beenden Sie die Sitzung des Dienstprogramms BTEQ.

Die nächste Task in dieser Tasksequenz ist die Überprüfung, ob die Teradata-Bibliothek für das Binden zur Laufzeit aktiviert ist.

### Zugehörige Tasks:

- „Hinzufügen von Teradata-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 415
- „Prüfen, ob die Teradata-Bibliothek für das Binden zur Laufzeit aktiviert ist (AIX)“ auf Seite 417

## Prüfen, ob die Teradata-Bibliothek für das Binden zur Laufzeit aktiviert ist (AIX)

Das Prüfen, ob die Teradata-Bibliothek für das Binden zur Laufzeit aktiviert ist, ist Teil der umfangreicheren Task des Hinzufügens von Teradata-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken.

Wenn Sie eine Teradata-Datenquelle zu einem Server mit zusammengeschlossenen Datenbanken unter AIX hinzufügen, müssen Sie vor der Registrierung von Wrappern oder Servern die Aktivierung des Bindens zur Laufzeit bestätigen.

### Vorgehensweise:

Gehen Sie wie folgt vor, um zu prüfen, ob die Teradata-Bibliothek für das Binden zur Laufzeit aktiviert ist:

1. Gehen Sie in das Verzeichnis, in dem sich die Datei libcliv2.so befindet.  
Während des Installationsprozesses wird diese Datei standardmäßig in das Verzeichnis /usr/lib platziert.
2. Geben Sie folgenden UNIX-Befehl aus.  
`dump -H libcliv2.so | grep libtli.a`
3. Prüfen Sie die Dateinamen, die auf der Anzeige erscheinen.  
Erscheint der Dateiname libtli.a, so ist die Teradata- Bibliothek für das Binden zur Laufzeit aktiviert.
4. Erscheint der Dateiname libtli.a nicht, geben Sie folgende UNIX-Befehle aus.  
`rtl_enable libcliv2.so -F libtli.a`  
`mv libcliv2.so libcliv2.so.old`  
`mv libcliv2.so.new libcliv2.so`  
`chmod a+r libcliv2.so`

Mit diesen Befehlen wird das Binden zur Laufzeit für die Teradata-Bibliothek aktiviert.

Die nächste Task in dieser Tasksequenz ist das Registrieren des Teradata-Wrappers.

#### Zugehörige Tasks:

- „Registrieren des Teradata-Wrappers“ auf Seite 421

## Einstellen der Teradata-Umgebungsvariablen

Beim Hinzufügen von Teradata zu einem Server mit zusammengeschlossenen Datenbanken müssen die Teradata-Umgebungsvariablen eingerichtet werden.

Wenn Sie DB2 Information Integrator installieren, wird während des Installationsprozesses versucht, die Teradata-Umgebungsvariablen in der Datei db2dj.ini einzurichten.

In folgenden Fällen werden die Umgebungsvariablen in der Datei db2dj.ini nicht eingerichtet:

- Die Teradata-Client-Software wird nach Einrichten des DB2-Servers mit zusammengeschlossenen Datenbanken installiert.
- Die Teradata-Client-Software wurde nicht installiert.

Die folgenden Umgebungsvariablen sind für Teradata gültig:

- COPERR
- COPLIB
- TERADATA\_CHARSET (optional)
- NETRACE (optional)
- COPANOMLOG (optional)

Die optionalen Umgebungsvariablen müssen manuell eingerichtet werden.

#### Voraussetzungen:

Ein Server mit zusammengeschlossenen Datenbanken, der ordnungsgemäß für den Zugriff auf Datenquellen eingerichtet ist. Dies umfasst die Installation und Konfiguration der erforderlichen Software wie beispielsweise der Client-Software von Datenquellen.



## Einschränkungen:

Siehe Abschnitt: Einschränkungen für die Datei db2dj.ini

## Vorgehensweise:

Um die erforderlichen Umgebungsvariablen automatisch einzurichten, gehen Sie wie folgt vor:

1. Installieren und konfigurieren Sie die Client-Software auf dem DB2-Server mit zusammengeschlossenen Datenbanken, sofern die Software nicht bereits installiert wurde.
2. Richten Sie die erforderlichen Umgebungsvariablen ein. Sie können die Umgebungsvariablen automatisch einrichten, indem Sie die Installation von DB2 Information Integrator erneut ausführen. Klicken Sie über die Klickstartleiste **Produkte installieren** an und befolgen Sie die Anweisungen des Assistenten.

Um die Umgebungsvariablen manuell einzurichten, gehen Sie wie folgt vor:

1. Editieren Sie die Datei db2dj.ini.
  - Auf Servern mit zusammengeschlossenen Datenbanken unter Windows befindet sich diese Datei im Verzeichnis sql1lib\cfg.
  - Auf Servern mit zusammengeschlossenen Datenbanken unter UNIX befindet sich diese Datei im Verzeichnis sql1lib/cfg.

Die Datei db2dj.ini enthält Konfigurationsdaten zu der auf dem Server mit zusammengeschlossenen Datenbanken installierten Teradata-Client-Software. Falls die Datei nicht vorhanden ist, können Sie mit Hilfe eines beliebigen Texteditors eine neue Datei mit dem Namen db2dj.ini erstellen. In der Datei db2dj.ini müssen Sie den vollständig qualifizierten Pfad für die Variablen angeben, da ansonsten Fehler auftreten.

2. Richten Sie je nach Bedarf die folgenden Umgebungsvariablen ein.

### COPERR

Setzen Sie die Umgebungsvariable COPERR auf den Pfad des Verzeichnisses, in dem die Datei errmsg.txt gespeichert ist. Geben Sie den vollständig qualifizierten Pfad für die Variable an: `COPERR=teradata-bibliotheksverzeichnis`. Beispiel:

```
COPERR=/usr/lib
```

### COPLIB

Setzen Sie die Umgebungsvariable COPLIB auf den Pfad des Verzeichnisses, in dem die Datei libcliv2.so gespeichert ist. Geben Sie den vollständig qualifizierten Pfad für die Variable an: `COPLIB=teradata-bibliotheksverzeichnis`. Beispiel:

```
COPLIB=/usr/lib
```

Die Dateien libcliv2.so und errmsg.txt befinden sich normalerweise in demselben Verzeichnis.

### TERADATA\_CHARSET

Wenn Sie die Variable TERADATA\_CHARSET nicht einrichten, ermittelt DB2 Information Integrator den Clientzeichensatz auf Grundlage der Codepage der Datenbank. Wird diese Variable eingerichtet, verwendet DB2 Information Integrator den Wert der Variablen als Clientzeichen-



satz. Der Wert der Variablen TERADATA\_CHARSET wird nicht geprüft. Wird jedoch ein falscher Wert angegeben, gibt die ferne Datenquelle einen Fehler zurück.

Setzen Sie die Umgebungsvariable TERADATA\_CHARSET in der Datei 'db2dj.ini' auf einen der folgenden gültigen Zeichensätze:

Auf Servern mit zusammengeschlossenen Datenbanken unter UNIX:

- HANGULKSC5601\_2R4
- KanjiEUC\_0U
- LATIN1\_0A
- LATIN9\_0A
- LATIN1252\_0A
- SCHGB2312\_1T0
- TCHBIG5\_1R0
- UTF8

Auf Servern mit zusammengeschlossenen Datenbanken unter Windows:

- HANGULKSC5601\_2R4
- KanjiSJIS\_0S
- LATIN1\_0A
- LATIN1252\_0A
- SCHGB2312\_1T0
- TCHBIG5\_1R0
- UTF8

3. Um sicherzustellen, dass die Umgebungsvariablen auf dem Server mit zusammengeschlossenen Datenbanken gesetzt werden, müssen Sie das DB2-Exemplar stoppen und erneut starten: Setzen Sie hierzu die folgenden Befehle ab:

```
db2stop  
db2start
```

#### **Aktivieren und Inaktivieren der Teradata-Tracefunktion:**

Die Teradata-Tracefunktion wird mit Hilfe der Teradata-Variablen NETRACE und COPANOMLOG aktiviert und inaktiviert. Sie können die Teradata-Tracefunktion aktivieren, wenn Sie eine Auflistung der an den Teradata-Server gesandten SQL-Anweisungen benötigen.

Um die Teradata-Tracefunktion zu aktivieren, editieren Sie die Datei db2dj.ini und verwenden Sie die folgenden Einstellungen für die relevanten Variablen:

```
NETRACE=1  
COPANOMLOG=tracedatei
```

tracedatei ist der vollständig qualifizierte Name der Datei, die die Tracedaten enthält. Diese Variablen aktivieren nur die Teradata-Tracefunktion; sie wirken sich nicht auf die DB2-Traceverarbeitung aus.

Um die Teradata-Tracefunktion zu inaktivieren, editieren Sie die Datei db2dj.ini und entfernen Sie sowohl die Variable NETRACE als auch die Variable COPANOMLOG.

Die nächste Task in dieser Tasksequenz ist das Registrieren des Teradata-Wrappers.

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Teradata-Tabellen und -Sichten“ auf Seite 428

**Zugehörige Referenzen:**

- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68
- „Einschränkungen für die Datei 'db2dj.ini'“ auf Seite 68

## Registrieren des Teradata-Wrappers

Die Registrierung des Teradata-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von Teradata-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.

Sie müssen einen Wrapper für den Zugriff auf Teradata-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

**Vorgehensweise:**

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER aus, und geben Sie den Standardnamen für den Wrapper an.

Beispiel:

```
CREATE WRAPPER TERADATA
```

**Empfehlung:** Verwenden Sie den Standardwrappernamen TERADATA. Registrieren Sie einen Wrapper mit dem Standardnamen, verwendet der Server mit zusammengeschlossenen Datenbanken automatisch den standardmäßigen Bibliotheksnamen, der diesem Wrapper zugeordnet ist.

Sollte zwischen dem Namen des Wrappers und einem bereits vorhandenen Wrappernamen in der zusammengeschlossenen Datenbank ein Namenskonflikt entstehen, können Sie den Standardnamen für den Wrapper durch einen beliebigen Namen ersetzen. Verwenden Sie einen anderen Namen als den Standardnamen, müssen Sie den Parameter LIBRARY in die Anweisung CREATE WRAPPER mit einschließen.

Um beispielsweise einen Wrapper mit dem Namen `tera_wrapper` auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER tera_wrapper LIBRARY 'libdb2teradata.a';
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung CREATE WRAPPER finden Sie in der Liste der Bibliotheksdateien von Teradata-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinition für den Teradata-Wrapper.

**Zugehörige Tasks:**

- „Registrieren der Serverdefinitionen für eine Teradata-Datenquelle“ auf Seite 422

#### Zugehörige Referenzen:

- „Teradata-Wrapper-Bibliotheksdateien“ auf Seite 422
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## Teradata-Wrapper-Bibliotheksdateien

Die folgende Tabelle enthält eine Liste der Verzeichnispfade und Bibliotheksdateinamen für den Teradata-Wrapper.

Wenn Sie DB2 Information Integrator installieren, werden dem in der Tabelle aufgeführten Verzeichnispfad 3 Bibliotheksdateien hinzugefügt. Wird der Server mit zusammengesetzten Datenbanken beispielsweise unter AIX ausgeführt, werden die Wrapperbibliotheksdateien `libdb2teradata.a`, `libdb2teradataF.a` und `libdb2teradataU.a` dem Verzeichnispfad hinzugefügt.

Wenn Sie einen Wrapper registrieren, geben Sie nur den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 86. Speicherpositionen und Dateinamen der Teradata-Wrapperbibliothek*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2teradata.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2teradata.sl
Solaris	/opt/IBM/db2/V8.1/lib	libdb2teradata.so
Windows	%DB2PATH%\bin	db2teradata.dll

`%DB2PATH%` ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows lautet `C:\Program Files\IBM\SQLLIB`.

#### Zugehörige Tasks:

- „Registrieren des Teradata-Wrappers“ auf Seite 421

## Registrieren der Serverdefinitionen für eine Teradata-Datenquelle

Die Registrierung der Serverdefinitionen für eine Teradata-Datenquelle ist Teil der umfangreicheren Task des Hinzufügens von Teradata-Datenquellen zu einem System zusammengesetzter Datenbanken.

In der zusammengesetzten Datenbank müssen Sie jeden Teradata-Server definieren, auf den Sie zugreifen möchten. Suchen Sie zuerst den Knotennamen der Teradata-Datenquelle, und verwenden Sie dann diesen Knotennamen bei der Registrierung des Servers.

#### Vorgehensweise:

Gehen Sie wie folgt vor, um eine Serverdefinition für eine Teradata-Datenquelle zu registrieren:

1. Suchen Sie den Knotennamen.
  - a. Suchen Sie die Hostdatei.

**Unter AIX-Betriebssystemen** ist die Hostdatei `/etc/hosts`.

**Unter Windows-Betriebssystemen** ist die Hostdatei

`x:\WINNT\system32\drivers\etc\hosts`. `x`: ist das Laufwerk, auf dem sich das Verzeichnis `\WINNT` befindet.

- b. Durchsuchen Sie die Hostdatei nach dem Aliasnamen des fernen Servers. Dieser Aliasname beginnt mit einer alphabetischen Zeichenfolge und endet mit dem Suffix `COP $n$` . Der Wert  $n$  ist die Nummer des Anwendungsprozessors, der dem Übertragungsprozessor von Teradata zugeordnet ist.
- c. Suchen Sie die Zeile in der Hostdatei, die den Aliasnamen enthält.
- d. Suchen Sie das erste nicht-numerische Feld in dieser Zeile.

**Beispiel für Hostdatei:**

```
127.0.0.1      localhost
9.22.5.77     nodexyz      nodexyzCOP1   # teradata server
9.66.111.133  rtplib05.data.xxx.com aap
9.66.111.161  rtpscm11.data.xxx.com aaprwt
9.66.111.161  rtpscm11.data.xxx.com accessm
```

In diesem Beispiel stellt das Feld **nodexyz** den Knotennamen dar.

2. Übergeben Sie die Anweisung **CREATE SERVER**.

Beispiel:

```
CREATE SERVER servername TYPE TERADATA VERSION 2.4 WRAPPER wrapper
      OPTIONS (NODE 'knotenname')
```

Sie müssen einen Servernamen angeben. Der angegebene Name muss eindeutig sein.

Der Parameter TYPE muss für alle Teradata-Server auf **TERADATA** gesetzt werden.

Der Teradata-Wrapper unterstützt alle Versionen von Teradata V2R3, V2R4 und V2R5. Geben Sie die Versionsnummer als zweistellige Zahl mit einem Dezimalzeichen an. Beispiele für gültige Versionsnummern sind 2.3, 2.4, 2.5.

Sie müssen einen Namen für den Wrapper angeben. Der angegebene Name muss einem Teradata-Wrapper entsprechen, den Sie mit der Anweisung **CREATE WRAPPER** registriert haben.

Außerdem ist die Angabe des Knotennamens, in dem sich der Teradata-Server befindet, erforderlich. Dieser Knotenname ist von der Groß-/Kleinschreibung abhängig.

Wenn Sie eine Teradata-Serverdefinition erstellen, können Sie in der Anweisung **CREATE SERVER** zusätzliche Serveroptionen angeben, falls erforderlich.

Nachdem Sie die Serverdefinition registriert haben, können Sie Serveroptionen durch Ausgeben der Anweisung **ALTER SERVER** hinzufügen oder löschen.

Die nächste Task in dieser Tasksequenz ist das Erstellen der Benutzerzuordnung für eine Teradata-Datenquelle.

**Zugehörige Tasks:**

- „Erstellen der Benutzerzuordnung für eine Teradata-Datenquelle“ auf Seite 425

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE SERVER - Beispiele für Teradata-Wrapper“ auf Seite 424

## Anweisung CREATE SERVER - Beispiele für Teradata-Wrapper

In diesem Abschnitt sind mehrere Beispiele für die Verwendung der Anweisung CREATE SERVER zur Registrierung von Servern für den Teradata-Wrapper enthalten. Der Abschnitt enthält ein vollständiges Beispiel für die Erstellung eines Servers mit allen erforderlichen Parametern sowie ein Beispiel zur Erstellung mit wahlfreien Serveroptionen.

### Vollständiges Beispiel:

Das nachfolgende Beispiel zeigt die Erstellung einer Serverdefinition für einen Teradata-Wrapper unter Verwendung der Anweisung CREATE SERVER:

```
| CREATE SERVER TERASERVER TYPE TERADATA  
|         VERSION 2.4 WRAPPER mein_wrapper  
|         OPTIONS (NODE 'tera-knoten');
```

Die Serveroption *TERASERVER* gibt den Namen an, den Sie dem Teradata-Server zuordnen. *TYPE TERADATA* gibt an, dass Sie den Zugriff auf eine Teradata-Datenquelle konfigurieren. *VERSION 2.4* ist die Version der Teradata-Server-Software, auf die Sie zugreifen möchten. *WRAPPER mein\_wrapper* gibt den Namen des Teradata-Wrappers an, den Sie mit der Anweisung CREATE WRAPPER registriert haben. *NODE 'tera-knoten'* ist der Name des Knotens, auf dem sich der Teradata-Server befindet.

### Beispiel mit Serveroptionen:

Das nachfolgende Beispiel zeigt eine Teradata-Serverdefinition mit Statistikdaten für das Optimierungsprogramm:

```
| CREATE SERVER TERASERVER1 TYPE TERADATA  
|         VERSION 2.4 WRAPPER WRAPPERNAME1  
|         OPTIONS (NODE 'tera-knoten1', CPU_RATIO '2.0', IO_RATIO '3.0');
```

In diesem Beispiel ist *TERASERVER1* der Name des Teradata-Servers, *WRAPPERNAME1* ist der Wrappernamen, den Sie mit der Anweisung CREATE WRAPPER registriert haben, und *'tera-knoten1'* ist der Name des Knotens, auf dem sich der Teradata-Server befindet. Die Serveroptionen *CPU\_RATIO* und *IO\_RATIO* stellen dem Optimierungsprogramm folgende Informationen zur Verfügung:

- Die CPU-Ressourcen des Servers mit zusammengeschlossenen Datenbanken verfügen im Vergleich zu den CPU-Ressourcen des Teradata-Servers über die doppelte Leistung.
- Die E/A-Einheiten des Servers mit zusammengeschlossenen Datenbanken verarbeiten Daten dreimal so schnell wie die E/A-Einheiten des Teradata-Servers.

### Zugehörige Tasks:

- „Registrieren der Serverdefinitionen für eine Teradata-Datenquelle“ auf Seite 422

### Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

## Erstellen der Benutzerzuordnung für eine Teradata-Datenquelle

Das Erstellen der Benutzerzuordnung für eine Teradata-Datenquelle ist Teil der umfangreicheren Task des Hinzufügens von Teradata-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken.

Wird versucht, auf einen Teradata-Server zuzugreifen, stellt der Server mit zusammengeschlossenen Datenbanken unter Verwendung einer gültigen Benutzer-ID mit Kennwort eine Verbindung zur Datenquelle her. Sie müssen eine Zuordnung zwischen der Benutzer-ID des Servers mit zusammengeschlossenen Datenbanken und dem zugehörigen Kennwort und der entsprechenden Benutzer-ID und dem Kennwort der Datenquelle definieren. Diese Zuordnung muss für alle Benutzer-IDs erstellt werden, die auf das System zusammengeschlossener Datenbanken zum Senden verteilter Anforderungen an die Teradata-Datenquelle zugreifen.

Sie müssen Benutzerzuordnungen für alle Teradata-Server erstellen, die Sie in der zugeordneten Anweisung CREATE SERVER registriert haben.

### Vorgehensweise:

Geben Sie eine Anweisung CREATE USER MAPPING aus, um die Benutzer-ID des Servers mit zusammengeschlossenen Datenbanken zu der Benutzer-ID und dem Kennwort des Teradata-Servers zuzuordnen.

Beispiel:

```
CREATE USER MAPPING FOR USERID SERVER TERASERVER
    OPTIONS (REMOTE_AUTHID 'remote_id', REMOTE_PASSWORD 'remote_password')
```

Alternativ können Sie Benutzerzuordnungen erstellen, indem Sie in der DB2-Steuerzentrale das Fenster zum Erstellen von Benutzerzuordnungen verwenden.

Die nächste Task in dieser Tasksequenz ist das Prüfen der Verbindung zwischen dem Server mit zusammengeschlossenen Datenbanken und dem Teradata-Server.

### Zugehörige Tasks:

- „Testen der Verbindung vom Server mit zusammengeschlossenen Datenbanken zum Teradata-Server“ auf Seite 426

### Zugehörige Referenzen:

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE USER MAPPING - Beispiele für Teradata-Wrapper“ auf Seite 425

## Anweisung CREATE USER MAPPING - Beispiele für Teradata-Wrapper

Dieser Abschnitt enthält Beispiele für die Verwendung der Anweisung CREATE USER MAPPING zur Zuordnung einer lokalen Benutzer-ID für den Server mit zusammengeschlossenen Datenbanken zu einer Benutzer-ID mit Kennwort für einen Teradata-Server. Dieser Abschnitt enthält ein vollständiges Beispiel mit allen erforderlichen Parametern sowie ein Beispiel für die Verwendung des DB2-Sonderregisters USER zusammen mit der Anweisung CREATE USER MAPPING.

### Vollständiges Beispiel:

Das nachfolgende Beispiel zeigt die Zuordnung einer lokalen Benutzer-ID für den Server mit zusammengeschlossenen Datenbanken (*MICHAEL*) zu einer Benutzer-ID mit Kennwort für einen Teradata-Server (*'mike'* und *'passxyz123'*):

```
CREATE USER MAPPING FOR MICHAEL SERVER TERASERVER
    OPTIONS (REMOTE_AUTHID 'mike', REMOTE_PASSWORD 'passxyz123')
```

Die Option *MICHAEL* gibt die Benutzer-ID für den Server mit zusammengeschlossenen Datenbanken an, die Sie zu einer für den Teradata-Server definierten Benutzer-ID zuordnen. *SERVER TERASERVER* gibt den Namen des Teradata-Servers an, den Sie in der Anweisung *CREATE SERVER* definiert haben. *REMOTE\_AUTHID 'mike'* ist die Benutzer-ID für den Teradata-Server, zu der Sie die lokale Benutzer-ID *MICHAEL* zuordnen. *REMOTE\_PASSWORD 'passxyz123'* ist das Kennwort, das dem Wert für *REMOTE\_AUTHID 'mike'* zugeordnet ist.

### Beispiel mit Sonderregister:

Das nachfolgende Beispiel zeigt eine Anweisung *CREATE USER MAPPING*, die das Sonderregister *USER* beinhaltet:

```
CREATE USER MAPPING FOR USER SERVER TERASERVER
    OPTIONS (REMOTE_AUTHID 'mike', REMOTE_PASSWORD 'passxyz123')
```

Sie können das DB2-Sonderregister *USER* dazu verwenden, die Berechtigungs-ID der Person, die die Anweisung *CREATE USER MAPPING* ausgibt, zur Berechtigungs-ID der Datenquelle zuzuordnen, die in der Benutzeroption *REMOTE\_AUTHID* angegeben wurde.

### Zugehörige Tasks:

- „Erstellen der Benutzerzuordnung für eine Teradata-Datenquelle“ auf Seite 425

### Zugehörige Referenzen:

- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

## Testen der Verbindung vom Server mit zusammengeschlossenen Datenbanken zum Teradata-Server

Das Testen der Verbindung vom Server mit zusammengeschlossenen Datenbanken zum Teradata-Server ist Teil der umfangreicheren Task des Hinzufügens von Teradata-Datenquellen zu Servern mit zusammengeschlossenen Datenbanken.

Sie können die Verbindung vom Server mit zusammengeschlossenen Datenbanken zum Teradata-Server durch Verwendung der von Ihnen festgelegten Serverdefinition und Benutzerzuordnung testen.

### Vorgehensweise:

Gehen Sie wie folgt vor, um die Verbindung zu testen:

1. Eröffnen Sie vom DB2-Befehlszeilenprozessor aus eine Durchgriffssitzung zum Ausgeben einer Anweisung *SELECT* für eine Teradata-Systemtabelle.

Beispiel:

```
SET PASSTHRU server_name
SELECT count(*) FROM dbc.tables
SET PASSTHRU RESET
```



Wenn die Anweisung SELECT eine Anzahl zurückgibt, sind Ihre Serverdefinition und Ihre Benutzerzuordnung ordnungsgemäß eingerichtet.

2. Gibt die Anweisung SELECT einen Fehler zurück, sind eventuell folgende Schritte erforderlich:
  - Stellen Sie sicher, dass der Teradata-Server für ankommende Verbindungen konfiguriert ist.
  - Prüfen Sie Ihre Benutzerzuordnung und stellen Sie sicher, dass die Einstellungen für die Optionen REMOTE\_AUTHID und REMOTE\_PASSWORD für Verbindungen zum Teradata-Server gültig sind. Ändern Sie die Benutzerzuordnung, oder erstellen Sie je nach Bedarf eine neue Benutzerzuordnung.
  - Stellen Sie sicher, dass die Teradata-Clientsoftware auf dem DB2-Server mit zusammengeschlossenen Datenbanken für die Verbindung zum Teradata-Server ordnungsgemäß installiert und konfiguriert ist.
  - Überprüfen Sie die Einstellungen Ihrer Variablen für den DB2-Server mit zusammengeschlossenen Datenbanken, um sicherzustellen, dass Sie auf den Teradata-Server zugreifen können. Zu diesen Variablen gehören die Systemumgebungsvariablen, die db2dj.ini-Variablen und die Variable der DB2-Profilregistrierdatenbank (db2set).
  - Überprüfen Sie Ihre Serverdefinition. Falls erforderlich, löschen Sie die Serverdefinition und erstellen Sie sie erneut.

Wenn Sie eine Durchgriffssitzung zum Ausgeben von SQL-Anweisungen für Teradata- Objekte eröffnen, können Sie keine Anweisung PREPARE mit einem Parameter INTO ausgeben, wenn die Anweisung Hostvariablen enthält.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für Teradata-Tabellen und -Sichten.

#### Zugehörige Tasks:

- „Hinzufügen von Teradata-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 415
- „Testen der Verbindung zum Teradata-Server“ auf Seite 416
- „Registrieren der Serverdefinitionen für eine Teradata-Datenquelle“ auf Seite 422
- „Registrieren von Kurznamen für Teradata-Tabellen und -Sichten“ auf Seite 428
- „Einstellen der Teradata-Umgebungsvariablen“ auf Seite 418

#### Zugehörige Referenzen:

- „ALTER USER MAPPING statement“ in *SQL Reference, Volume 2*

## Teradata-Kurznamen auf Servern mit zusammengeschlossenen Datenbanken

Sie müssen auf allen Teradata-Servern, die Sie definiert haben, für jede Teradata<sup>®</sup>-Tabelle und -Sicht, auf die Sie zugreifen möchten, einen Kurznamen erstellen. Verwenden Sie bei der Abfrage von Teradata-Servern diese Kurznamen anstatt der Namen der Datenquellenobjekte.

Der Server mit zusammengeschlossenen Datenbanken stellt unter Verwendung des Kurznamens, den Sie mit der Anweisung CREATE NICKNAME zugewiesen haben, eine Verbindung zu der Teradata-Datenquelle her. Der Server mit zusammengeschlossenen Datenbanken fragt dann den Datenquellenkatalog ab und überprüft die Verbindung zur Datenquelle. Kann die Verbindung nicht hergestellt werden, generiert DB2<sup>®</sup> eine Fehlernachricht.



Die zusammengeschlossene Datenbank verwendet Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Diese Statistiken werden gesammelt, wenn Sie einen Kurznamen für ein Datenquellenobjekt erstellen.

Die zusammengeschlossene Datenbank stellt das Vorhandensein des Objekts in der Datenquelle sicher und versucht dann, bestehende statistische Daten von dieser Datenquelle zu sammeln. Informationen, die für das Optimierungsprogramm nützlich sind, werden aus den Katalogen der Datenquelle gelesen und in den globalen Katalog auf dem Server mit zusammengeschlossenen Datenbanken aufgenommen. Da einige oder alle der Kataloginformationen der Datenquelle vom Optimierungsprogramm verwendet werden können, ist es ratsam, Statistiken in der Datenquelle zu aktualisieren, bevor ein Kurzname erstellt wird. Aktualisieren Sie diese Statistiken in der Datenquelle durch Verwendung eines Befehls oder Dienstprogramms, der/das äquivalent ist zu dem DB2-Befehl **RUNSTATS**.

Eine SQL-Anweisung **INSERT**, **UPDATE** oder **DELETE** kann nicht an einen Kurznamen übergeben werden, der auf eine aktualisierbare Teradata-Sicht verweist, es sei denn, diese SQL-Anweisung kann vollständig an die Teradata-Datenquelle ausgelagert werden.

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Teradata-Tabellen und -Sichten“ auf Seite 428

**Zugehörige Referenzen:**

- „**RUNSTATS** Command“ in *Command Reference*
- „Anweisung **CREATE NICKNAME** - Beispiele für den Teradata-Wrapper“ auf Seite 429

## Registrieren von Kurznamen für Teradata-Tabellen und -Sichten

Die Registrierung der Kurznamen für Teradata-Tabellen und -Sichten ist Teil der umfangreicheren Task des Hinzufügens von Teradata-Datenquellen zu einem System zusammengeschlossener Datenbanken.

Registrieren Sie für jeden Teradata-Server, den Sie definieren, einen Kurznamen für jede Tabelle und Sicht, auf die Sie zugreifen möchten.

**Vorgehensweise:**

Übergeben Sie zum Registrieren eines Kurznamens die Anweisung **CREATE NICKNAME**.

Beispiel:

```
CREATE NICKNAME TERANICKNAME FOR TERASERVER."remote_schema"."remote.table"
```

**Empfehlung:** Da die zusammengeschlossene Datenbank Katalogstatistiken für Objekte mit Kurznamen zur Optimierung der Abfrageverarbeitung einsetzt, sollten Sie die Statistikdaten in der Teradata-Datenquelle aktualisieren, bevor Sie einen Kurznamen registrieren. Sie können einen Befehl oder ein Dienstprogramm verwenden, der/das äquivalent ist zu dem DB2-Befehl **RUNSTATS**.

Kurznamen können bis zu 128 Zeichen lang sein.

Sie können beim Ausgeben der Anweisung CREATE NICKNAME die Spaltenoption NUMERIC\_STRING angeben. Diese Spaltenoption können Sie außerdem auch bei Verwendung der Anweisung ALTER NICKNAME angeben.

**Zugehörige Konzepte:**

- „Teradata-Kurznamen auf Servern mit zusammengeschlossenen Datenbanken“ auf Seite 427

**Zugehörige Referenzen:**

- „RUNSTATS Command“ in *Command Reference*
- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den Teradata-Wrapper“ auf Seite 429

## Anweisung CREATE NICKNAME - Beispiele für den Teradata-Wrapper

Dieser Abschnitt enthält ein Beispiel für die Verwendung der Anweisung CREATE NICKNAME zur Registrierung eines Kurznamens für eine Teradata-Tabelle oder -Sicht, auf die Sie zugreifen möchten.

Dieses Beispiel zeigt die Angabe eines fernen Objekts für den Teradata-Server, zu dem der Kurzname zugeordnet wird:

```
CREATE NICKNAME TERASALES FOR TERASERVER."salesdata"."europe"
```

TERASALES ist der eindeutige Kurzname, den Sie der Teradata-Tabelle oder -Sicht zuordnen. Ein Kurzname besteht aus zwei Teilen: dem Schema und dem eigentlichen Kurznamen. Übergehen Sie das Schema beim Erstellen des Kurznamens, erstellt DB2 den Kurznamen unter Verwendung Ihrer Berechtigungs-ID als Schema.

TERASERVER."salesdata"."europe" gibt eine dreiteilige Kennung für das ferne Objekt an:

- TERASERVER ist der Name, den Sie in der Anweisung CREATE SERVER der Teradata-Datenbank zugeordnet haben.
- salesdata ist der Name des fernen Schemas, zu dem die Tabelle oder Sicht gehört.
- europe ist der Name der fernen Tabelle oder Sicht, auf die Sie zugreifen möchten.

**Zugehörige Konzepte:**

- „Teradata-Kurznamen auf Servern mit zusammengeschlossenen Datenbanken“ auf Seite 427

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Teradata-Tabellen und -Sichten“ auf Seite 428

**Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

---

## Optimierung und Fehlerbehebung der Konfiguration für Teradata-Datenquellen

Wenn Sie die Konfiguration für Teradata-Datenquellen eingerichtet haben, müssen Sie sie möglicherweise ändern, um die Leistung zu verbessern und potenzielle Fehler zu beheben.

### Fehler bei UPDATE- oder DELETE-Operationen für Kurznamen

Zeilen für Tabellen von Teradata-Datenquellen werden standardmäßig nicht eindeutig angegeben. Möglicherweise empfangen Sie einen Fehler mit dem Fehlercode SQL30090N, RC="21", wenn Sie versuchen, einen Kurznamen zu aktualisieren oder zu löschen, der einer Teradata-Tabelle oder -Sicht zugeordnet ist. Tritt ein solcher Fehler (SQL30090N, RC="21") auf, erstellen Sie mindestens einen eindeutigen Index für die Teradata-Tabelle, die aktualisiert oder gelöscht wird, und versuchen Sie erneut, die Operation auszuführen.

### Optimierung und Inaktivierung der Teradata-Zugriffprotokollierung

Das Produkt Teradata stellt eine Funktion für die Zugriffprotokollierung bereit. Diese generiert Protokolleinträge, wenn Teradata die spezifischen Sicherheitszugriffsrechte verschiedener Benutzer für eine oder mehrere Datenbanken überprüft. Die Zugriffprotokollierung stellt eine beachtliche Menge an aussagefähigen Sicherheitsinformationen bereit. Diese Funktion erhöht die Prozessorbelegung jedoch erheblich und kann die Systemleistung beeinträchtigen.

Sollte eine Verbesserung der Systemleistung erforderlich sein, werten Sie die Regeln für die Überprüfungsberechtigungen aus, die Sie für die Zugriffprotokollierung definiert haben. Beenden Sie danach alle nicht erforderlichen Regeln durch die Definierung von Anweisungen END LOGGING.

Die Leistung ist optimal, wenn Sie die Zugriffprotokollierung ganz ausschalten. Löschen Sie das Makro **Teradata DBC.AccLogRules**, und erzwingen Sie anschließend ein Zurücksetzen der anerkannten Parallelanwendung (TPA = Trusted Parallel Application), um die Zugriffprotokollierung vollständig zu stoppen.

Weitere Informationen finden Sie in der Teradata-Dokumentation.

### Aktivieren des Bindens zur Laufzeit für libcliv2.so (AIX)

Beim Ausführen der Datei `djxlinkTeradata.sh` zum Verbinden mit der gemeinsam benutzten Teradata-Bibliothek namens `libcliv2.so` erhalten Sie beim Ausgeben der Anweisung `CREATE NICKNAME` möglicherweise eine Fehlermeldung.

Ein Beispiel einer solchen Fehlermeldung ist:

```
DB21034E:  
Der Befehl wurde als SQL-Anweisung verarbeitet, da es sich um keinen gültigen Befehl  
des Befehlszeilenprozessors handelte. Während der SQL-Verarbeitung wurde Folgendes  
ausgegeben:  
SQL30081N Kommunikationsfehler. Verwendetes Kommunikationsprotokoll:  
"TCP/IP". Verwendete Kommunikations-API: "SOCKETS". Position, an der der Fehler  
festgestellt wurde: "9.112.26.28". Kommunikationsfunktion, die den Fehler  
feststellte: "recv". Protokollspezifische(r) Fehlercode(s): "*", "*", "0".  
SQLSTATE=08001
```

Erhalten Sie eine Fehlermeldung, überprüfen Sie das Verzeichnis /sql lib/db2dump auf Trapdateien. Trapdateien beginnen mit dem Buchstaben t und enden mit einem Suffix 000. Beispiel:

```
t123456.000
```

Überprüfen Sie die Trace-Information in der Trapdatei auf Verweise auf die Funktion `OsCall`, die angeben, dass diese Funktion das Stoppen des Servers mit zusammengeschlossenen Datenbanken verursacht hat.

Das nachfolgende Beispiel zeigt Trace-Informationen mit dem Verweis auf eine Funktion `OsCall`, der möglicherweise in einer Trapdatei enthalten sein kann:

**\*\*\* Start stack traceback \*\*\***

```
0x239690E0 OsCall + 0x28C
0x23973FB0 mtdpassn + 0x8A4
0x239795A4 mtdp + 0x208
0x2395A928 MTDPIO + 0x28C
0x239609C4 CLICON + 0xD50
0x23962350 DBCHCL + 0xC4
```

Ist in einer der Trapdateien ein Verweis auf eine Funktion `OsCall` enthalten, geben Sie die folgenden UNIX-Befehle aus:

```
cd /usr/lib
rtl_enable libcliv2.so -F libtli.a
mv libcliv2.so libcliv2.so.old
mv libcliv2.so.new libcliv2.so
chmod a+r libcliv2.so
```

Diese Befehle aktivieren das Binden zur Laufzeit für die gemeinsam benutzte Bibliothek `libcliv2.so`.

#### **Zugehörige Tasks:**

- „Hinzufügen von Teradata-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 415
- „Prüfen, ob die Teradata-Bibliothek für das Binden zur Laufzeit aktiviert ist (AIX)“ auf Seite 417

#### **Zugehörige Referenzen:**

- „db2set - DB2 Profile Registry Command“ in *Command Reference*



## Kapitel 23. Konfigurieren des Zugriffs auf Web-Services-Datenquellen

Die Informationen in diesem Abschnitt erläutern, wie Web-Services-Datenquellen Ihrem System zusammengeschlossener Datenbanken hinzugefügt werden.

### Der Web-Services-Wrapper und das WSDL-Dokument

Web-Services-Provider werden durch WSDL-Dokumente beschrieben (WSDL = Web Services Description Language, Sprache zur Beschreibung von Web-Services). Mit Hilfe des Web-Services-Wrappers können Sie auf Web-Services-Provider zugreifen. Wie aus Abb. 28 ersichtlich wird, implementiert der Web-Services-Provider einen Service und veröffentlicht die Schnittstelle für einen Service-Broker wie beispielsweise UDDI. Der Serviceanforderer kann anschließend mit Hilfe des Service-Brokers nach einem Web-Service suchen. Wenn der Anforderer einen Service findet, stellt er eine Bindung zum Service-Provider her, sodass der Anforderer den Web-Service nutzen kann. Der Anforderer ruft den Service auf, indem er SOAP-Nachrichten (SOAP = Simple Object Access Protocol) mit dem Provider austauscht.

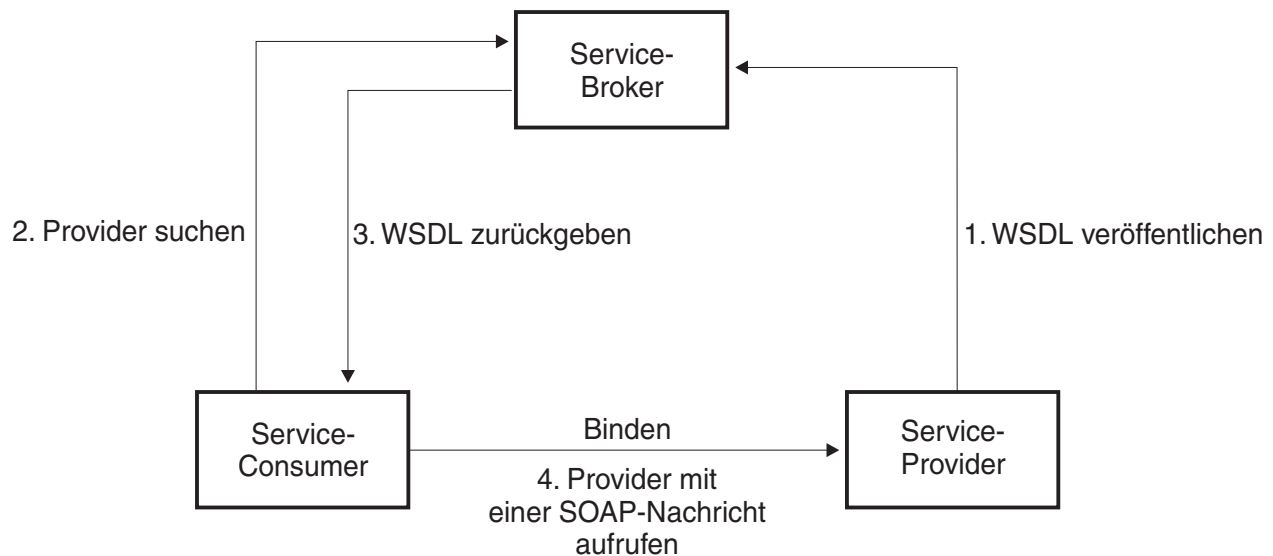


Abbildung 28. Web-Services: Eine serviceorientierte Architektur

Die SOAP-Spezifikation definiert das Layout einer XML-basierten Nachricht. Eine SOAP-Nachricht ist in einer SOAP-Rahmenanweisung enthalten. Die Rahmenanweisung besteht aus optionalen SOAP-Kopfdaten (Header) und einem obligatorischen SOAP-Hauptteil. Der SOAP-Header kann Informationen zur Nachricht enthalten, wie beispielsweise Verschlüsselungsinformationen und Authentifizierungsinformationen. Der SOAP-Hauptteil enthält die Nachricht. Die SOAP-Spezifikation definiert außerdem eine Standardverschlüsselung für Programmiersprachenbindungen, die als SOAP-Verschlüsselung bezeichnet wird.

| Der Schlüssel zum Web-Service ist das WSDL-Dokument. Das WSDL-Dokument ist  
| ein XML-Dokument, das Web-Services anhand von gesendeten und empfangenen  
| Nachrichten beschreibt. Die Nachrichten werden mit Hilfe eines Systems von  
| Typen beschrieben, bei dem es sich normalerweise um das XML-Schema handelt.  
| Bei einer Web-Services-Operation wird mindestens einer Nachricht ein Nach-  
| richtenaustauschmuster zugeordnet. Ein Nachrichtenaustauschmuster identifiziert  
| die Reihenfolge und Kardinalität von gesendeten und empfangenen Nachrichten  
| sowie den logischen Empfänger bzw. Sender, an den die Nachrichten gesendet  
| bzw. von dem die Nachrichten empfangen werden. Eine Schnittstelle  
| fasst die Operationen zu einer Gruppe zusammen, ohne dass eine Festlegung auf  
| das Übertragungsformat oder physische Format erfolgt. Detaillierte Angaben zum  
| Übertragungsformat und physischen Format für mindestens eine Schnittstelle wer-  
| den in einer WSDL-Bindung gemacht. Ein Endpunkt ordnet einer Bindung eine  
| Netzadresse zu. Ein Service fasst Endpunkte, die eine gemeinsame Schnittstelle  
| implementieren, zu einer Gruppe zusammen. Die Nachrichten können dokumen-  
| tenorientierte Informationen oder prozessorientierte Informationen enthalten, die  
| auch als Fernprozeduraufrufe (Remote Procedure Calls, RPC) bezeichnet werden.  
| Ein WSDL-Dokument kann einen Web-Service oder mehrere Web-Services enthal-  
| ten.

| Das Beispiel in Abb. 29 auf Seite 435 zeigt die WSDL-Definition eines einfachen  
| Service, der Börsennotierungen zur Verfügung stellt. Der Web-Service unterstützt  
| eine einzige Operation mit dem Namen 'GetLastTradePrice'. Zugriff auf den Ser-  
| vice besteht mit dem Protokoll SOAP 1.1 über HTTP. Die Anforderung liest ein  
| Tickersymbol (tickerSymbol) als Eingabe vom Datentyp STRING und gibt die  
| Kursnotierung (price) als Datentyp FLOAT zurück. Bei den Datentypen STRING  
| und FLOAT handelt es sich um vordefinierte Typen in den XML-Schemastan-  
| dards. Ein Web-Service kann ebenfalls Datentypen definieren und diese benutzer-  
| definierten Datentypen in Nachrichten verwenden. Vordefinierte und benutzer-  
| definierte XML-Datentypen werden für Spalten der Kurznamen zugeordnet. Das  
| vollständige Beispiel und die WSDL-Spezifikation finden Sie auf der W3C-Website.  
|

---

```

<?xml version="1.0"?>
<definitions name="StockQuote"
...

<types>
  <schema targetNamespace="http://example.com/stockquote.xsd"
    xmlns="http://www.w3.org/2000/10/XMLSchema">
    <element name="TradePriceRequest">
      <complexType>
        <all>
          <element name="tickerSymbol" type="string"/>
        </all>
      </complexType>
    </element>
    <element name="TradePrice">
      <complexType>
        <all>
          <element name="price" type="float"/>
        </all>
      </complexType>
    </element>
  </schema>
</types>

<message name="GetLastTradePriceInput">
...
</message>

  <portType name="StockQuotePortType">
    <operation name="GetLastTradePrice">
      <input message="tns:GetLastTradePriceInput"/>
      <output message="tns:GetLastTradePriceOutput"/>
    </operation>
  </portType>

  <binding name="StockQuoteSoapBinding"
    type="tns:StockQuotePortType">
    <soap:binding style="document"
      transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="GetLastTradePrice">
      <soap:operation soapAction="http://example.com/GetLastTradePrice"/>
      <input>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
    </operation>
  </binding>

  <service name="StockQuoteService">
    <documentation>My first service</documentation>
    <port name="StockQuotePort" binding="tns:StockQuoteBinding">
      <soap:address location="http://example.com/stockquote"/>
    </port>
  </service>
</definitions>

```

---

Abbildung 29. Beispiel eines WSDL-Dokuments

Der Web-Services-Wrapper verwendet die Operationen in einem Anschlussstyp (portType), der über eine SOAP-Bindung mit einem HTTP-Transport verfügt. Die



Eingabenachrichten in der Operation und die zugeordneten Typen oder Elemente werden zu Spalten im Kurznamen. Die Ausgabenachrichten in der Operation werden in die Kurznamenhierarchie extrahiert. Für jede Operation im WSDL-Dokument können Sie eine eigene Kurznamenhierarchie erstellen.

Abb. 30 zeigt ein WSDL-Dokument, das einen Anschlusstyp (portType) mit dem Operationsnamen GETTEMP enthält. Bei diesem Web-Service geben Sie eine Postleitzahl (zipcode) als Eingabe ein und erhalten eine Temperaturangabe für den der Postleitzahl entsprechenden Ort.

```
<?xml version="1.0"?>
<definitions name="TemperatureService" targetNamespace=http://www.xmethods.net/
  sd/TemperatureService.wsdl"
  xmlns:tns="http://www.xmethods.net/sd/TemperatureService.wsdl"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns="http://schemas.xmlsoap.org/wsdl/">
  <message name="getTempRequest">
    <part name="zipcode" type="xsd:string"/>
  </message>
  <message name="getTempResponse">
    <part name="return" type="xsd:float"/>
  </message>
  <portType name="TemperaturePortType">
    <operation name="getTemp">
      <input message="tns:getTempRequest"/>
      <output message="tns:getTempResponse"/>
    </operation>
  </portType>
  <binding name="TemperatureBinding" type="tns:TemperaturePortType">
    <soap:binding style="rpc"
      transport="http://schemas.xmlsoap.org/soap/http" />
    <operation name="getTemp">
      <soap:operation soapAction="" />
      <input>
        <soap:body use="encoded" namespace="urn:xmethods-Temperature"
          encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
      </input>
      <output>
        <soap:body use="encoded" namespace="urn:xmethods-Temperature"
          encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
      </output>
    </operation>
  </binding>
  <service name="TemperatureService">
    <documentation>
      Returns current temperature in a given U.S. zipcode
    </documentation>
    <port name="TemperaturePort" binding="tns:TemperatureBinding">
      <soap:address
        location="http://services.xmethods.net:80/soap/servlet/rpcrouter" />
    </port>
  </service>
</definitions>
```

Abbildung 30. Web-Service GETTEMP

Der Eingabewert wird von der Tabelle 'zipcode' beschrieben. Der Ausgabewert wird von der Rückkehrspalte (return) beschrieben. Im WSDL-Dokument werden diese Spalten im Nachrichtenelement (messages) identifiziert. Das Nachrichtenelement stellt die logische Definition der Daten dar, die zwischen dem Web-Services-Provider und dem Web-Services-Consumer gesendet werden.

Werden weitergehende Erklärungen der Informationen im Nachrichtenelement benötigt, kann das WSDL-Dokument auch ein Typelement (type) enthalten. Das Typelement kann auf vordefinierte Typen verweisen, die auf den XML-Schemaspezifikationen basieren, oder auf Typen, die von einem Benutzer definiert wurden.

Abb. 31 zeigt den Kurznamen, den das Discovery-Tool der DB2<sup>®</sup>-Steuerzentrale aus dem WSDL-Dokument generiert. Die Spalte 'zipcode' ist aufgrund der Kurznamensyntax von TEMPLATE eine erforderliche Eingabespalte:

---

```
CREATE NICKNAME GETTEMP (
  ZIPCODE VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  RETURN VARCHAR (48) OPTIONS(XPATH './return/text()')
)
FOR SERVER "EHPWSSERV"
OPTIONS(URL 'http://services.xmethods.net:80/soap/servlet/rpcrouter',
  SOAPACTION ' ',
  TEMPLATE '<soapenv:Envelope>
    <soapenv:Body>
      <ns2:getTemp>
        <zipcode>&zipcode[1,1]</zipcode>
      </ns2:getTemp>
    </soapenv:Body>
  </soapenv:Envelope>',
  XPATH '/soapenv:Envelope/soapenv:Body/*',
  NAMESPACES ' ns1="http://www.xmethods.net/sd/TemperatureService.wsdl",
              ns2="urn:xmethods-Temperature" ,
              soapenv="http://schemas.xmlsoap.org/soap/envelope/" );
```

---

Abbildung 31. Kurzname GETTEMP

Die Kurznamenoptionen URL und SOAPACTION des Web-Services-Wrappers ermöglichen es, den Endpunkt oder die beim Erstellen des Kurznamens angegebene Adresse zu überschreiben. Wenn Sie die für URLCOLUMN oder SOAPACTIONCOLUMN aktivierten Spalten in einer Abfrage verwenden, können Sie dynamische Adressen mit denselben Kurznamen verwenden. Wenn Sie die Kurznamenoptionen URL und SOAPACTION beim Erstellen eines Kurznamens definieren und URLCOLUMN und SOAPACTIONCOLUMN für die Spaltenoption aktivieren, werden die Late Binding-Funktionen von Web-Services-Wrappern verwendet. Der Wert für die Kurznamenoption SOAPACTION wird zu einem Attribut im HTTP-Header. Der Wert für die Kurznamenoption URL ist die HTTP-URL, an die die Abfrage gesendet wird.

Die Kurznamenoptionen URL und SOAPACTION stellen dynamische Kurznamen-zuordnungen zur Verfügung. Diese dynamischen Adressen sind nützlich, wenn mehrere Unternehmen einen Web-Service-Anschlusstyp (portType) implementieren. Der Web-Services-Wrapper erfordert, dass die einzigen Unterschiede zwischen den WSDL-Dokumenten verschiedene URL- und SOAPACTION-Werte sind. Sie können die Late Binding-Funktion verwenden, um denselben Kurznamen für verschiedene Service-Endpunkte zu erstellen und zu verwenden, die verschiedene Unternehmen möglicherweise verwenden wollen. Die Werte für URL und SOAPACTION werden aus dem WSDL-Dokument abgeleitet.

Das folgende Beispiel zeigt, wie Sie die Spaltenoptionen URLCOLUMN und SOAPACTIONCOLUMN verwenden können:

---

```

CREATE NICKNAME GetPartQuote(
  partnumber INTEGER OPTIONS (TEMPLATE'&column'),
  price FLOAT OPTIONS (XPATH './price'),
  urlcol VARCHAR(100) OPTIONS (URLCOLUMN 'Y'),
  soapactioncol VARCHAR(100) OPTIONS (SOAPACTIONCOLUMN 'Y'),
  FOR SERVER myServer
  OPTIONS (
    ...
    SOAPACTION 'http://example.com/GetPartPrice' ,
    URL 'http://mycompany.com:9080/GetPartPrice',
    ...
  )

```

---

Abbildung 32. Kurzname 'GetPartQuote'

Im folgenden Beispiel werden die Spalten URLCOL und SOAPACTIONCOL verwendet, die mit aktivierter Spaltenoption URLCOLUMN und aktivierter Spaltenoption SOAPACTIONCOLUMN definiert wurden:

```

SELECT * FROM supplier_endpoints p,
       GetPartQuote q
WHERE partnumber=1234 AND
       p.url=q.urlcol AND
       p.soapaction=q.soapactioncol;

```

Die SQL-Anwendung kann die Auswahl der zu verwendenden Endpunkte (endpoints) so lange verzögern, bis eine Abfrage ausgeführt wird, anstatt einen spezifischen Endpunkt zu dem Zeitpunkt zu definieren, an dem der Kurzname erstellt wird.

Der Web-Services-Wrapper kann ein großes Volumen an WSDL-Dokumentdaten in Fragmente aufteilen, um die Gesamtmenge des verwendeten Speichers zu reduzieren. Geben Sie in der DB2-Steuerzentrale auf der Seite 'Einstellungen' des Fensters 'Merkmale' die Option **STREAMING** an, wenn Sie einen Web-Services-Kurznamen erstellen. Der Web-Services-Wrapper verarbeitet den daraus resultierenden XML-Datenstrom und extrahiert anschließend die von einem Abfragefragment angeforderten Informationen. Der Web-Services-Wrapper führt die Syntaxanalyse immer für ein Fragment zur Zeit aus. Daher sollte die Option **STREAMING** nur zur syntaktischen Analyse großer XML-Dokumente verwendet werden.

**Zugehörige Konzepte:**

- „WSDL from a DADX file“ in *IBM DB2 Information Integrator Application Developer's Guide*
- „Web services description language“ in *IBM DB2 Information Integrator Application Developer's Guide*

**Zugehörige Tasks:**

- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammenschlossenen Datenbanken“ auf Seite 439
- „Registrieren des Web-Services-Wrappers“ auf Seite 440
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442

**Zugehörige Referenzen:**

- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Nachrichten für den Web-Services-Wrapper“ auf Seite 468

---

## Hinzufügen von Web-Services zu einem System zusammengeschlossener Datenbanken

### Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken

Um den Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Web-Services-Datenquellen zu konfigurieren, müssen Sie dem Server Informationen zu den Datenquellen und -objekten zur Verfügung stellen, auf die Sie zugreifen möchten. Bei diesen Informationen kann es sich beispielsweise um ein gültiges WSDS-Dokument handeln (WSDS = Web Services Description Language, Sprache zur Beschreibung von Web-Services).

Der Zugriff des Servers mit zusammengeschlossenen Datenbanken auf Web-Services-Datenquellen kann über die DB2-Steuerzentrale oder die DB2-Befehlszeile konfiguriert werden. Die DB2-Steuerzentrale umfasst einen Assistenten, der Sie durch die erforderlichen Schritte zum Konfigurieren des Servers mit zusammengeschlossenen Datenbanken führt.

#### Voraussetzungen:

- DB2 Information Integrator muss auf einem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss auf dem Server mit zusammengeschlossenen Datenbanken vorhanden sein.

#### Vorgehensweise:

Um Web-Services-Datenquellen einem Server mit zusammengeschlossenen Datenbanken hinzuzufügen, gehen Sie wie folgt vor:

1. Registrieren Sie den Wrapper.
2. Registrieren Sie die Serverdefinition.
3. Optional: Erstellen Sie eine Benutzerzuordnung.
4. Registrieren Sie Kurznamen für die Web-Services-Datenquellen.
5. Optional: Erstellen Sie Sichten für zusammengeschlossene Datenbanken für die Web-Services-Kurznamen.

#### Zugehörige Konzepte:

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Web-Services-Wrapper - Sicherheit“ auf Seite 467
- „Web-Services-Wrapper - Sicherheit“ auf Seite 467
- „Der Web-Services-Wrapper und das WSDL-Dokument“ auf Seite 433

#### Zugehörige Tasks:

- „Editieren der Oracle-Prozedur 'genclntsh' und Erstellen der Datei 'libclntsh' nach der Installation von DB2 Information Integrator (HP-UX, Linux, Solaris)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Registrieren des Web-Services-Wrappers“ auf Seite 440
- „Registrieren der Serverdefinition für Web-Services-Datenquellen“ auf Seite 441
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442
- „Erstellen von Sichten für zusammengeschlossene Datenbanken für Web-Services-Kurznamen“ auf Seite 450

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Registrieren des Web-Services-Wrappers“ auf Seite 440

**Zugehörige Referenzen:**

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Nachrichten für den Web-Services-Wrapper“ auf Seite 468
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172

## Registrieren des Web-Services-Wrappers

Beim Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken muss der Web-Services-Wrapper registriert werden.

Die Registrierung eines Wrappers ist für den Zugriff auf Web-Services-Datenquellen erforderlich. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken verwendet, um mit Datenquellen zu kommunizieren und Daten aus Datenquellen abzurufen. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

**Vorgehensweise:**

Um einen Wrapper zu registrieren, setzen Sie eine Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Wrapperbibliotheksdatei ab.

Um beispielsweise einen Wrapper mit dem Namen websr\_wrapper auf dem Server mit zusammengeschlossenen Datenbanken zu registrieren, der das Betriebssystem Windows verwendet, setzen Sie die folgende Anweisung ab:

```
CREATE WRAPPER websr_wrapper LIBRARY 'db2ws.d11';
```

Der Name der von Ihnen angegebenen Wrapperbibliotheksdatei hängt von dem Betriebssystem auf dem Server mit zusammengeschlossenen Datenbanken ab. Den korrekten Namen, der in der Anweisung CREATE WRAPPER angegeben werden muss, finden Sie in der Liste der Bibliotheksdateien für den Web-Services-Wrapper. Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinitionen für den Web-Services-Wrapper.

**Zugehörige Konzepte:**

- „Web-Services-Wrapper - Sicherheit“ auf Seite 467
- „Der Web-Services-Wrapper und das WSDL-Dokument“ auf Seite 433

**Zugehörige Tasks:**

- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 439
- „Registrieren der Serverdefinition für Web-Services-Datenquellen“ auf Seite 441

**Zugehörige Referenzen:**

- „Bibliotheksdateien des Web-Services-Wrappers“ auf Seite 441

## Bibliotheksdateien des Web-Services-Wrappers

Die folgende Tabelle enthält eine Liste der Verzeichnispfade und Bibliotheksdateinamen für den Web-Services-Wrapper.

Wenn Sie DB2 Information Integrator installieren, werden dem in der Tabelle aufgeführten Verzeichnispfad 3 Bibliotheksdateien hinzugefügt. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden die Wrapperbibliotheksdateien `libdb2ws.a`, `libdb2wsF.a` und `libdb2wsU.a` dem Verzeichnispfad hinzugefügt.

Wenn Sie einen Wrapper registrieren, geben Sie nur den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 87. Speicherpositionen und Dateinamen der Bibliothek des Web-Services-Wrappers*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2ws.a
Windows	%DB2PATH%\bin	db2ws.dll

%DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist. Der Standardverzeichnispfad unter Windows lautet `C:\Program Files\IBM\SQLLIB`.

### Zugehörige Tasks:

- „Registrieren des Web-Services-Wrappers“ auf Seite 440

## Registrieren der Serverdefinition für Web-Services-Datenquellen

Beim Hinzufügen von Web-Services zu einem System zusammengeschlossener Datenbanken muss die Serverdefinition für eine Web-Services-Datenquelle registriert werden.

Nach der Registrierung des Wrappers muss ein entsprechender Server registriert werden. Für jeden Web-Service, auf den Sie zugreifen wollen, muss jeweils eine Serverdefinition registriert werden.

Sie können die Serverdefinition über eine DB2-Befehlszeile oder die DB2-Steuerzentrale registrieren.

### Vorgehensweise:

Um eine Serverdefinition im System zusammengeschlossener Datenbanken für den Web-Services-Wrapper zu registrieren, setzen Sie die Anweisung `CREATE SERVER` ab.

Um beispielsweise unter Windows eine Serverdefinition namens `ws_server` für Web-Services zu registrieren, setzen Sie die folgende Anweisung ab:

```
CREATE SERVER ws_server WRAPPER websr_wrapper;
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für die Web-Services-Datenquellen.

#### **Zugehörige Tasks:**

- „Registrieren des Web-Services-Wrappers“ auf Seite 440
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442

#### **Zugehörige Referenzen:**

- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172

## **Registrieren von Kurznamen für Web-Services-Datenquellen**

### **Registrieren von Kurznamen für Web-Services-Datenquellen**

Beim Hinzufügen von Web-Services zu einem System zusammenschlossener Datenbanken müssen Kurznamen für Web-Services-Datenquellen registriert werden.

Für jede Web-Services-Operation wird jeweils eine Kurznamenhierarchie erstellt. Die Web-Services-Operationen werden im WSDL-Dokument (WSDL = Web Services Description Language) definiert. Übergeordnete Kurznamen enthalten mindestens einen untergeordneten Kurznamen. Untergeordnete Kurznamen entsprechen den Elementen, die innerhalb des Elements für den übergeordneten Kurznamen verschachtelt sind.

Sie können den Kurznamen über eine DB2-Befehlszeile oder über die DB2-Steuerzentrale erstellen.

In der DB2-Steuerzentrale können Sie das Discovery-Tool verwenden, um die Kurznamen schnell zu erstellen. Die Eingabe für das Discovery-Tool ist eine URL der Speicherposition eines WSDL-Dokuments. Das Discovery-Tool erstellt Kurznamen als Ergebnis der Verarbeitung des WSDL-Dokuments. Das WSDL-Dokument kann Schemadefinitionen enthalten, die entweder in die WSDL-Datei integriert sind oder in eine externe XML-Schemadatei, die in die WSDL-Datei importiert wird. Diese Schemadefinitionen werden mit Hilfe von URL-Adressen importiert.

#### **Voraussetzungen:**

Sie müssen auf ein gültiges WSDL-Dokument zugreifen können, das den Web-Service beschreibt, mit dem Sie kommunizieren wollen.

#### **Einschränkungen:**

- Es werden lediglich Anforderungs- und Antwortoperationen unterstützt.
- Als Bindung wird lediglich eine SOAP-Bindung mit einem HTTP-Transport unterstützt.
- Für jede Spalte müssen Sie entweder die Option TEMPLATE oder die Option XPATH verwenden. Dies gilt nicht für die Spalten mit den Optionen SOAPACTIONCOLUMN, URLCOLUMN, PRIMARY\_KEY oder FOREIGN\_KEY.



## Vorgehensweise:

Um Kurznamen für Web-Services-Datenquellen über die DB2-Befehlszeile zu erstellen, setzen Sie eine Anweisung CREATE NICKNAME ab.

Um beispielsweise unter Windows die Kurznamen für einen Web-Service namens GETTEMP zu registrieren, setzen Sie die folgende Anweisung ab:

```
CREATE NICKNAME GETTEMP (  
  ZIPCODE VARCHAR (48) OPTIONS(TEMPLATE '&column'),  
  RETURN VARCHAR (48) OPTIONS(XPATH './return/text()')  
)  
FOR SERVER "EHPWSSERV"  
OPTIONS(URL 'http://services.xmethods.net:80/soap/servlet/rpcrouter',  
  SOAPACTION ' ' ,  
  TEMPLATE '<soapenv:Envelope>  
    <soapenv:Body>  
      <ns2:getTemp>  
        <zipcode>&zipcode[1,1]</zipcode>  
      </ns2:getTemp>  
    </soapenv:Body>  
  </soapenv:Envelope>',  
  XPATH '/soapenv:Envelope/soapenv:Body/*' ,  
  NAMESPACES ' ns1="http://www.xmethods.net/sd/TemperatureService.wsdl",  
    ns2="urn:xmethods-Temperature" ,  
    soapenv="http://schemas.xmlsoap.org/soap/envelope/"');
```

Die Datendefinitionssprache (Data Definition Language, DDL), die von der DB2-Steuerzentrale generiert wird, ordnet alle Eingabelemente für Spalten des Rootkurznamens in der Kurznamenhierarchie zu. Der erstellte Kurzname wird aus dem WSDL-Dokument abgeleitet.

Um Kurznamen für Web-Services-Datenquellen über die DB2-Steuerzentrale zu registrieren, gehen Sie wie folgt vor:

1. Erweitern Sie den Ordner **Objekte zusammenschlossener Datenbanken**.
2. Erweitern Sie den Ordner des Wrappers, für den Sie Kurznamen registrieren wollen.
3. Erweitern Sie den Ordner **Serverdefinitionen**.
4. Erweitern Sie den Ordner des Servers, für den Sie Kurznamen registrieren wollen.
5. Klicken Sie den Ordner **Kurznamen** mit Maustaste 2 an und wählen Sie **Erstellen** aus.
6. Klicken Sie im Fenster 'Kurznamen erstellen' die Option **Aufspüren** an, um Suchkriterien für die Auswahl von Objekten in der Datenquelle zu definieren.
7. Geben Sie das WSDL-Dokument an, das die Definition des Web-Service enthält, auf den Benutzer von DB2 Information Integrator Zugriff haben sollen. Das WSDL-Dokument kann entweder lokal vorhanden sein, oder Sie können die Speicherposition des Dokuments mit Hilfe einer URL angeben.
8. Klicken Sie **OK** an, um den Kurznamen gemäß dem ausgewählten WSDL-Dokument zu erstellen. Die DB2-Steuerzentrale extrahiert das WSDL-Dokument in mehrere DDL-Anweisungen zum Erstellen von Kurznamen und verwendet hierbei die entsprechenden Definitionen für die Abhängigkeiten zwischen übergeordneten und untergeordneten Elementen. Die erstellten Kurznamen stellen die Web-Services-Hierarchie dar, die im WSDL-Dokument definiert ist.

Optional: Die nächste Task in dieser Tasksequenz ist das Erstellen von Sichten für zusammenschlossene Datenbanken für die Web-Services-Kurznamen.



### Zugehörige Konzepte:

- „Web-Services-Wrapper - Sicherheit“ auf Seite 467

### Zugehörige Tasks:

- „Angaben von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75
- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 439
- „Registrieren des Web-Services-Wrappers“ auf Seite 440

### Zugehörige Referenzen:

- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Nachrichten für den Web-Services-Wrapper“ auf Seite 468
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172

## Die Option TEMPLATE auf der Kurznamen- und Spaltenebene

Dieser Abschnitt bezieht sich auf den WebSphere® Business Integration-Wrapper und den Web-Services-Wrapper.

Der WebSphere® Business Integration-Wrapper und der Web-Services-Wrapper erstellen XML-Dokumente, die für den WebSphere Business Integration-Adapter und die Web-Services-Umgebung erforderlich sind. Zum Zeitpunkt der Erstellung des Kurznamens benötigen die Wrapper die Schablonenfragmente auf Kurznamen- und Spaltenebene. Dies ist die Option TEMPLATE in der Anweisung CREATE NICKNAME. Die Wrapper verwenden diese Informationen während der Abfrageplanungsphase und der Abfrageausführungsphase.

### Web-Services-Wrapper

Beim Web-Services-Wrapper variieren die erforderlichen und optionalen Attribute gemäß den Definitionen im WSDL-Dokument und gemäß der Art und Weise, wie eine Spalte abgeleitet wird. Eine Spalte kann entweder von einem Element oder von einem Attribut eines Elements abgeleitet werden.

- Wird die Spalte von einem Element abgeleitet, wird mit dem Wert für das Attribut 'minOccurs' festgelegt, ob eine Spalte optional oder erforderlich ist.
- Ist für 'minOccurs' der Wert 0 angegeben, ist die Spalte optional.
- Ist für 'minOccurs' der Wert 1 angegeben, ist die Spalte erforderlich.
- Wird die Spalte von einem Attribut eines Elements abgeleitet, wird mit dem Wert für 'use' im Attribut festgelegt, ob die Spalte optional oder erforderlich ist.
- Enthält ein Attribut den Wert 'use=optional', ist die Spalte optional.
- Enthält ein Attribut den Wert 'use=required', ist die Spalte erforderlich.

Das folgende Beispiel zeigt ein Attribut in einer Schemadefinition, die einer Spalte zugeordnet ist:

```
<xsd:complexType>
  <xsd:sequence>
    <xsd:element ref="tns:ZooName"/>
    <xsd:element ref="tns:Count"/>
    <xsd:element ref="tns:LastModified"/>
```

```

    <xsd:element maxOccurs="unbounded" minOccurs="0" ref="tns:Zookeeper"/>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:string" use="optional"/>
</xsd:complexType>

```

## WebSphere Business Integration-Wrapper

Beim WebSphere Business Integration-Wrapper variieren die erforderlichen und optionalen Spalten gemäß der Anwendung und dem zugeordneten Adapter. Sie müssen die erforderlichen und optionalen Eingabespalten identifizieren, indem Sie für diese Spalten die entsprechenden Werte für die Schablonenoption TEMPLATE angeben. Bevor Sie die Kurznamen mit Hilfe der DB2<sup>®</sup>-Steuerzentrale erstellen, müssen Sie die XML-Schemadefinitionsdatei modifizieren, um die erforderlichen und optionalen Eingabespalten entsprechend zu markieren.

### SAP-BAPI

Die IBM<sup>®</sup> DB2-Steuerzentrale legt die erforderlichen und optionalen Eingabespalten auf Grundlage des Wertes bestimmter Markierungen in den XML-Schemadefinitionsdateien (XSD-Dateien) fest, die die Geschäftsobjektdefinition darstellen.

Im Anmerknungsabschnitt ('annotation') eines Elements auf einer beliebigen Ebene der Geschäftsobjekthierarchie (übergeordnete oder untergeordnete Geschäftsobjekte) gibt das Präfix I im Wert für 'appSpecificInfo' einen Importparameter für die SAP-BAPI an, der die Geschäftsobjektdefinition zugeordnet ist. Das Präfix E gibt einen Exportparameter für die SAP-BAPI an. Einige Elemente können sowohl Import- als auch Exportparameter für eine BAPI sein. Das folgende Beispiel zeigt ein Element, bei dem es sich sowohl um einen Import- als auch einen Exportparameter handelt:

```
<bx:appSpecificInfo>ICOMPANYCODE:ECOMPANYCODE</bx:appSpecificInfo>
```

Die Präfixe werden vom ODA-Tool (Object Discovery Agent = Objekterkennungsagent) von WebSphere Business Integration automatisch generiert. Grundlage für die Generierung sind die Informationen, die aus dem SAP-Geschäftsobjektrepository extrahiert werden.

Wird ein Element, das einen Importparameter darstellt (Präfix I im Wert für 'appSpecificInfo'), mit dem Attribut minOccurs=1 angegeben, identifiziert die DB2-Steuerzentrale das Element als erforderlichen Eingabeparameter und markiert die Elemente in der Kurznamendefinition als erforderliche Eingabespalte. Das ODA-Tool von WebSphere Business Integration setzt den Wert von minOccurs für die erforderlichen Eingabeparameter der SAP-BAPI nicht automatisch auf 1. Sie müssen auf das SAP-Geschäftsobjektrepository verweisen, um alle erforderlichen Eingabeparameter für die BAPI zu ermitteln, auf die Sie zugreifen wollen. Anschließend müssen Sie die entsprechenden Elemente in der XML-Schemadatei editieren, indem Sie das Attribut minOccurs=1 manuell setzen. Wird für einen Eingabeparameter weiterhin die Standardeinstellung 0 für das Attribut minOccurs verwendet, gibt die DB2-Steuerzentrale die Spalte in der generierten Kurznamenhierarchie als optionale Eingabespalte an.

Das folgende Beispiel zeigt eine optionale Eingabespalte:

```

<xsd:element name="Company_code" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>ICOMPANYCODE:</bx:appSpecificInfo>

```

```

        <bx:attributeInfo isForeignKey="false" isKey="true" />
    </bx:boAttribute>
</xsd:appinfo>
</xsd:annotation>
<xsd:simpleType>
    <xsd:restriction base="xsd:string">
        <xsd:maxLength value="4" />
    </xsd:restriction>
</xsd:simpleType>
</xsd:element>

```

Das folgende Beispiel zeigt eine erforderliche Eingabespalte:

```

<xsd:element name="Company_id" minOccurs="1">
    <xsd:annotation>
        <xsd:appinfo>
            <bx:boAttribute>
                <bx:appSpecificInfo>ICOMPANYID:</bx:appSpecificInfo>
                <bx:attributeInfo isForeignKey="true" isKey="true" />
            </bx:boAttribute>
        </xsd:appinfo>
    </xsd:annotation>
    <xsd:simpleType>
        <xsd:restriction base="xsd:string">
            <xsd:maxLength value="4" />
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>

```

Die erforderlichen und optionalen Eingabespalten für SAP-Geschäftsanwendungen werden durch die in folgender Tabelle dargestellte Syntax ausgewiesen:

*Tabelle 88. Markierungsschema für SAP-Eingabespalteninformationen*

Verwendete Markierungen in SAP-XSD-Dateien	Erforderliche Eingabespalte	Spaltenverweis in Kurznamenschemata
Alle Elemente auf beliebiger Hierarchieebene mit Präfix = 'I' und 'minOccurs=1'	Ja	&spaltenname[1,1]
Alle Elemente auf beliebiger Hierarchieebene mit Präfix = 'O' und 'minOccurs=0'	Nein	&spaltenname[0,1]

### Siebel und PeopleSoft

Die DB2-Steuerzentrale legt die erforderlichen und optionalen Eingabespalten auf Grundlage des Vorhandenseins und des Wertes der Markierung `isRequired` im Abschnitt `attributeInfo` der Anmerkung `annotation` für das Element fest. Ist keine Markierung `isRequired` vorhanden, ist die Spalte keine erforderliche Eingabespalte. Das ODA-Tool von WebSphere Business Integration generiert diese Markierungen in der XSD-Datei nicht automatisch. Sie müssen die erforderlichen und optionalen Eingabespalten ermitteln und sie in der XSD-Datei entsprechend markieren, bevor Sie mit Hilfe der DB2-Steuerzentrale die Kurznamen-DDL generieren.

Das folgende Beispiel zeigt die Markierungen für eine erforderliche Eingabespalte und für optionale Eingabespalten in der XSD-Datei für eine Siebel- bzw. PeopleSoft-Geschäftsobjektdefinition.

---

```

<xsd:element name="sieb_ssa_Contact_Contact">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boDefinition version="1.0.0">
        <bx:appSpecificInfo>ON=Contact;CN=Contact</bx:appSpecificInfo>
      </bx:boDefinition>
    </xsd:appinfo>
  </xsd:annotation>
<xsd:complexType>
  <xsd:sequence>
<xsd:element name="Id" minOccurs="0">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>FN=Id</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false"
          isKey="true" isRequired="true" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...
</xsd:element>
...

```

---

Abbildung 33. Abschnitt einer Siebel-Geschäftsobjektdefinition (Teil 1 von 2)

---

```

...
<xsd:element name="FirstName" minOccurs="1">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>FN=First Name</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false"
          isRequired="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
<xsd:simpleType>
  <xsd:restriction base="xsd:string">
    <xsd:maxLength value="50" />
  </xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="LastName" minOccurs="1">
  <xsd:annotation>
    <xsd:appinfo>
      <bx:boAttribute>
        <bx:appSpecificInfo>FN=Last Name</bx:appSpecificInfo>
        <bx:attributeInfo isForeignKey="false" isKey="false"
          isRequired="false" />
      </bx:boAttribute>
    </xsd:appinfo>
  </xsd:annotation>
  ...

```

---

Abbildung 33. Abschnitt einer Siebel-Geschäftsobjektdefinition (Teil 2 von 2)

Die erforderlichen und optionalen Eingabespalten für Siebel- und PeopleSoft-Geschäftsanwendungen werden durch die in folgender Tabelle dargestellte Syntax ausgewiesen:

*Tabelle 89. Markierungsschema für Siebel- und PeopleSoft-Eingabespalteninformationen*

Verwendete Markierungen in Siebel- und PeopleSoft-XSD-Dateien	Erforderliche Eingabespalte	Spaltenverweis in Kurznamen-schablone
isRequired="true"	Ja	&spaltenname[1,1]
isRequired="false"	Nein	&spaltenname[0,1]

Das folgende Beispiel zeigt die DDL, die von der DB2-Steuerzentrale auf Grundlage der XSD-Datei erstellt wird, die in der Abbildung mit der Bezeichnung *Abschnitt einer Siebel-Geschäftsobjektdefinition* dargestellt ist. Die XSD-Datei in dieser Abbildung enthält für das Attribut 'isRequired' den Wert 'false'.

```
CREATE NICKNAME sieb_ssa_contact_contact_nn(
  Id VARCHAR(15) OPTIONS(XPATH './ns1:Id/text()'),
    TEMPLATE '<ns1:Id>&column</ns1:Id>'),
  FirstName VARCHAR(50) OPTIONS(XPATH './ns1:FirstName/text()'),
    TEMPLATE '<ns1:FirstName>&column</ns1:FirstName>'),
  LastName VARCHAR(50) OPTIONS(XPATH './ns1:LastName/text()'),
    TEMPLATE '<ns1:LastName>&column</ns1:LastName>'),
  AccountId VARCHAR(255) OPTIONS(XPATH './ns1:AccountId/text()'),
  PrimaryAccountName VARCHAR(100)
    OPTIONS(XPATH './ns1:PrimaryAccountName/text()'),
  PrimaryPostalCode VARCHAR(30)
    OPTIONS(XPATH './ns1:PrimaryPostalCode/text()'),
  PrimaryStreetAddress VARCHAR(200)
    OPTIONS(XPATH './ns1:PrimaryStreetAddress/text()'),
  SalesRep VARCHAR(255) OPTIONS(XPATH './ns1:SalesRep/text()'),
  State VARCHAR(255) OPTIONS(XPATH './ns1:State/text()')
  FOR SERVER siebel_server
  OPTIONS(XPATH '//ns1:sieb_ssa_contact_contact',
    TEMPLATE '<ns1:sieb_ssa_contact_contact>
      &Id[1,1] &FirstName[0,1] &LastName[0,1]
    </ns1:sieb_ssa_contact_contact>',
    BUSOBJ_NAME 'sieb_ssa_contact_contact',
    NAMESPACES 'ns1="http://www.ibm.com/websphere/
      crossworlds/2002/BOSchema/sieb_ssa_contact_contact"');
```

#### Zugehörige Konzepte:

- „Der WebSphere Business Integration-Wrapper“ auf Seite 137
- „Der Web-Services-Wrapper und das WSDL-Dokument“ auf Seite 433

#### Zugehörige Tasks:

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengesetzter Datenbanken“ auf Seite 144
- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 439
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442

### Zugehörige Referenzen:

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175
- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158
- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Nachrichten für den Web-Services-Wrapper“ auf Seite 468
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172

### Kurznamen und XPATH-Ausdrücke

Dieser Abschnitt bezieht sich auf den WebSphere® Business Integration-Wrapper und den Web-Services-Wrapper.

Kurznamen entsprechen der Baumstruktur der Daten Ihres XML-Dokuments. Übergeordnete und untergeordnete Kurznamen (Eltern und Kinder) entsprechen der Stammstruktur (Root) und den verschachtelten Elementen der Datenbaumstruktur. Diese übergeordneten und untergeordneten Kurznamen werden durch Primär- und Fremdschlüssel miteinander verbunden, die in der Anweisung CREATE NICKNAME angegeben werden.

Jeder Kurzname wird durch XPath-Ausdrücke definiert, die Ausgabewerte darstellen. Der WebSphere Business Integration-Wrapper und der Web-Services-Wrapper verwendet XPath-Ausdrücke, um eine Entsprechung zwischen den Daten in einem XML-Dokument und den Zeilen in einer relationalen Tabelle herzustellen. Diese XPath-Ausdrücke identifizieren die Werte im XML-Dokument und ermitteln, welche Entsprechung diese Werte zu den Spalten der einzelnen Zeilen haben. Die Daten im XML-Dokument werden vom WebSphere Business Integration-Wrapper und vom Web-Services-Wrapper lediglich gelesen. Die Wrapper führen keine Aktualisierung der Daten durch. Die Option XPATH enthält die Informationen zum Ermitteln der SOAP-Nachrichten über die Kennungen der SOAP-Rahmenanweisung und des SOAP-Hauptteils. Die Nachricht 'getQuote' ist in den Elementen der SOAP-Rahmenanweisung und des SOAP-Hauptteils enthalten.

Der XPATH-Ausdruck der Option NICKNAME zeigt auf sich wiederholende Kennungen, die sich im Ausgabeelement befinden. Der XPath-Ausdruck legt fest, wie viele bzw. welche Zeilen der Kurzname enthalten wird. Der XPATH-Ausdruck der Spaltenoption steht in einem relativen Verhältnis zum XPATH-Ausdruck NICKNAME. Der XPATH-Ausdruck der Spaltenoption identifiziert die Werte in einer Zeile. Ein XPATH-Ausdruck der Option NICKNAME in einem untergeordneten Kurznamen steht in einem relativen Verhältnis zu einem XPATH-Ausdruck der Option NICKNAME in einem übergeordneten Kurznamen.

Beim Erstellen eines Kurznamens werden Optionen ausgewählt, die die Beziehung zwischen dem Kurznamen und dem XML-Dokument angeben. Kurznamen, die für WebSphere Business Integration-Wrapper erstellt werden, sind einem XML-Schemadefinitionsdokument (XSD-Dokument) zugeordnet. Kurznamen, die für Web-Services-Wrapper erstellt werden, sind einem WSDL-Dokument zugeordnet.

### **Zugehörige Konzepte:**

- „Was ist XML?“ auf Seite 471
- „Der Web-Services-Wrapper und das WSDL-Dokument“ auf Seite 433
- „Datenzuordnungen zwischen Kurznamen und XML-Dokumenten“ auf Seite 479

### **Zugehörige Tasks:**

- „Hinzufügen von XML zu einem System zusammenschlossener Datenbanken“ auf Seite 474
- „Registrieren von Kurznamen für XML-Datenquellen“ auf Seite 482
- „Erstellen von Sichten für zusammenschlossene Datenbanken für Nicht-Stammkurznamen (XML-Wrapper)“ auf Seite 487
- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammenschlossener Datenbanken“ auf Seite 144
- „Registrieren von Kurznamen für Geschäftsanwendungsdatenquellen“ auf Seite 148
- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammenschlossenen Datenbanken“ auf Seite 439
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442

### **Zugehörige Referenzen:**

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175
- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158
- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172
- „Anweisung CREATE NICKNAME - Beispiele für den XML-Wrapper“ auf Seite 482

## **Erstellen von Sichten für zusammenschlossene Datenbanken für Web-Services-Kurznamen**

Beim Hinzufügen von Web-Services-Datenquellen zu einem System zusammenschlossener Datenbanken können Sichten für zusammenschlossene Datenbanken für Web-Services-Kurznamen erstellt werden.

Sie können Sichten für zusammenschlossene Datenbanken für die Hierarchie der Kurznamen definieren, die ein Web-Services-Dokument beschreiben. Durch das Definieren von Sichten für zusammenschlossene Datenbanken wird sichergestellt, dass die Abfragen, die Teile einer Hierarchie von Web-Services-Kurznamen verknüpfen, ordnungsgemäß ausgeführt werden können.

### **Vorgehensweise:**

Um Sichten für zusammenschlossene Datenbanken für Web-Services-Kurznamen zu definieren, gehen Sie wie folgt vor:

1. Definieren Sie eine Sicht für mindestens einen Web-Services-Kurznamen. Wenn Sie alle Kurznamen verknüpfen wollen, die zu einer Operation im Web-Service gehören, müssen Sie eine Sicht definieren, die alle diese Kurznamen einschließt.



2. Verwenden Sie in der Klausel WHERE der Sicht Verknüpfungsprädikate für alle Spalten, die durch die Spaltenoptionen PRIMARY\_KEY und FOREIGN\_KEY zusammengehören.

In nachstehendem Beispiel befindet sich der Primärschlüssel in der Spalte ooport\_getzooreport\_pkey im Kurznamen zooport\_getzooreport\_report\_nn. Der Fremdschlüssel befindet sich in der Spalte ooport\_getzooreport\_fkey im Kurznamen zooport\_getzooreport\_report\_report\_nn.

```
CREATE VIEW zooreport
(zoooid, zooname, number_of_zookeeper,
lastmodified, zookeeper_id, zookeeper_name,
fingers_left, animal_name, animal_species, animal_lot)
AS ( SELECT zoooid, report_zooname,
report_count, report_lastmodified,
zookeeper_id, zk.report_name, report_numberfingersleft,
a.report_name, report_species,
report_lot
FROM zooport_getzooreport_nn ,
zooport_getzooreport_report_nn as zk,
zooport_getzooreport_report_report_nn as a
WHERE zk.ooport_getzooreport_pkey=a.ooport_getzooreport_fkey
AND zooport_getzooreport_pkey=zk.ooport_getzooreport_fkey);
```

Mit der folgenden Anweisung SELECT können Sie Informationen aus allen Kurznamen abrufen:

```
SELECT * FROM zooreport WHERE zoooid='1';
```

Dies war die letzte Task in dieser Tasksequenz.

#### Zugehörige Referenzen:

- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172

### Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper

Wenn Sie einen Kurznamen für den Zugriff auf einen Web-Service erstellen, erstellen Sie jeweils eine Eingabespalte für jeden Wert in der Eingabenachricht einer Web-Services-Operation und jeweils eine Ausgabespalte für jeden Wert in der Ausgabenachricht einer Web-Services-Operation. Die Definitionen der Eingabe- und Ausgabespalten werden mit den Definitionen der Kurznamenspaltenoptionen gesteuert.

Die Spaltenoption TEMPLATE gibt an, dass es sich bei einer Spalte um eine Eingabespalte handelt. Die Spaltenoption XPATH gibt an, dass es sich bei einer Spalte um eine Ausgabespalte handelt. Wenn die Kurznamenspaltenoption TEMPLATE die in eckige Klammern eingeschlossene Notation [1,1] enthält, ist die Spalte eine erforderliche Eingabespalte. Enthält die Kurznamenspaltenoption TEMPLATE die in eckige Klammern eingeschlossene Notation [0,1], ist die Spalte eine optionale Eingabespalte.

Die Kurznamenspaltenoption NAMESPACES ist eine Liste mit durch Kommata getrennten Name/Wert-Paaren, die ein System zusammenschlossener Datenbanken verwendet, um die Namensbereiche aufzulösen, die für Elemente in XML-Eingabe- und



XML-Ausgabedokumenten verwendet werden. Die Namensbereiche werden in der Nachrichtenanforderung verwendet, um die in der Kurznamenoption TEMPLATE verwendeten Präfixe zu definieren. Die Kurznamenoption NAMESPACES wird verwendet, um die Präfixe, die in XPath-Ausdrücken verwendet werden, mit den URIs von Namensbereichen aufzulösen, die in den WSDL- oder XML-Schemata definiert sind. Die XPath-Ausdrücke werden auf das XML-Dokument angewendet, das vom Web-Service zurückgegeben wird.

### Beispiel 1: Erforderliche Eingabespalten

Das folgende Beispiel zeigt einen Kurznamen, der einen Web-Service namens 'getQuote' verwendet. Der Web-Service liest ein Börsentickersymbol und gibt eine Kursnotierung zurück. Die folgende DDL wird vom Discovery-Tool in der DB2-Steuerzentrale erstellt:

```
CREATE NICKNAME "stockquote.stockquoteport_getquote_nn" (  
    symbol VARCHAR (48) OPTIONS(TEMPLATE '&column'),  
    result VARCHAR (48) OPTIONS(XPATH './Result/text()'))  
FOR SERVER "xmethods_server" OPTIONS(  
    URL 'http://66.28.98.121:9090/soap' ,  
    SOAPACTION 'urn:xmethods-delayed-quotes#getQuote' ,  
    TEMPLATE '<soapenv:Envelope>  
        <soapenv:Body>  
            <ns2:getQuote>  
                <symbol>&symbol [1,1]</symbol>  
            </ns2:getQuote>  
        </soapenv:Body>  
    </soapenv:Envelope>',  
    XPATH '/soapenv:Envelope/soapenv:Body/*' ,  
    NAMESPACES 'ns2="urn:xmethods-delayed-quotes" ,  
                ns1="http://www.theminelectric.com/wsdl/  
                net.xmethods.services.stockquote.StockQuote/" ,  
                soapenv="http://schemas.xmlsoap.org/soap/envelope/" ');
```

Die Kurznamenoption TEMPLATE gibt die Spalte SYMBOL als erforderliche Eingabespalte an, da sie die Bezeichnung [1,1] enthält. In der Kurznamenoption TEMPLATE wird die vollständige SOAP-Rahmenanweisung (Envelope) für den Web-Service angegeben. Der Eingabewert für 'getQuote' ist in der SOAP-Rahmenanweisung und den Hauptteilelementen (Body) enthalten. Die Kurznamenoption XPATH enthält die Informationen zum Ermitteln der Kursnotierungswerts über die Kennungen der SOAP-Rahmenanweisung und der Hauptteilelemente.

Verwenden Sie den Kurznamen "stockquote.stockquoteport\_getquote\_nn", um auf den Web-Service zuzugreifen, wie in der folgenden Abfrage dargestellt:

```
SELECT * FROM "stockquote.stockquoteport_getquote_nn"  
WHERE symbol='IBM';
```

Sie müssen in dieser Anweisung das Vergleichselement symbol='IBM' verwenden, da 'symbol' eine erforderliche Eingabespalte ist. Das Gleichheitszeichen (=) ist das einzig gültige Vergleichselement für Eingabespalten. Jedes dieser Vergleichselemente mit Gleichheitszeichen setzt einen Wert in der Eingabenachricht. Ist die Eingabespalte optional, ist für diese Spalte kein Gleichheitszeichen als Vergleichselement erforderlich. Ist die Eingabespalte erforderlich, muss die Abfrage mit einem Gleichheitszeichen als Vergleichselement abgesetzt werden. Sie können einen Literalwert wie beispielsweise IBM in einem Ausdruck oder Wert mit Gleichheitszeichen aus einer verknüpften Tabelle oder einem verknüpften Kurznamen verwenden.

## Beispiel 2: Wiederholung von Elementen und untergeordneten Kurznamen

Das folgende Beispiel verwendet einen Web-Service namens `getZooReport`, der einen Bericht für Zoos (`zooreport`) erstellt. Der Eingabewert ist eine Zoo-ID. Der Ausgabewert ist ein Bericht, der durch das folgende Schema beschrieben wird:

```
<wsdl:definitions name="Name"
  targetNamespace="http://myzoo.com"
  ...
<wsdl:types>
  <xsd:schema elementFormDefault="qualified" targetNamespace="http://myzoo.com"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:element name="Animal">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tns:Name"/>
          <xsd:element ref="tns:Species"/>
          <xsd:element ref="tns:Lot"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="AnimalCareList">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element maxOccurs="unbounded" minOccurs="1" ref="tns:Animal"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="Count" type="xsd:string"/>
    <xsd:element name="LastModified" type="xsd:string"/>
    <xsd:element name="Name" type="xsd:string"/>
    <xsd:element name="NumberFingersLeft" type="xsd:string"/>
    <xsd:element name="Species" type="xsd:string"/>
    <xsd:element name="Zoo">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tns:ZooName"/>
          <xsd:element ref="tns:Count"/>
          <xsd:element ref="tns:LastModified"/>
          <xsd:element maxOccurs="unbounded" minOccurs="0" ref="tns:Zookeeper"/>
        </xsd:sequence>
        <xsd:attribute name="id" type="xsd:string" use="optional"/>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="ZooName" type="xsd:string"/>
    <xsd:element name="Zookeeper">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element ref="tns:Name"/>
          <xsd:element ref="tns:NumberFingersLeft"/>
          <xsd:element ref="tns:AnimalCareList"/>
        </xsd:sequence>
        <xsd:attribute name="id" type="xsd:string" use="optional"/>
      </xsd:complexType>
    </xsd:element>
  </xsd:schema>
</wsdl:types>
  ...
```

Abbildung 34. Web-Service 'getZooReport'

Es folgt die DDL, die vom Discovery-Tool der DB2-Steuerzentrale generiert wird.  
Grundlage hierfür ist die WSDL-Datei, die das Schema enthält.

---

```
CREATE NICKNAME zooport_getzooreport_nn (
  zooid VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  zoo_id VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/@ns1:id'),
  report_zooname VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/ns1:ZooName/text()'),
  report_count VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/ns1:Count/text()'),
  report_lastmodified VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:LastModified/text()'),
  zooport_getzooreport_pkey VARCHAR (16) FOR BIT DATA NOT NULL
    OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER "zooserver" OPTIONS(
  URL 'http://localhost:9080/MaelstromTest/services/ZooPort' ,
  SOAPACTION 'http://myzoo.com/getZooReport' ,
  TEMPLATE '<soapenv:Envelope>
    <soapenv:Body>
      <zooId>&zooId[1,1]</zooId>
    </soapenv:Body>
  </soapenv:Envelope>' ,
  XPATH '/soapenv:Envelope/soapenv:Body' ,
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
              ns1="http://myzoo.com " ');
```

---

Abbildung 35. Zoobericht (zooreport) – übergeordneter Kurzname – zooport\_getzooreport\_nn

---

```
CREATE NICKNAME zooport_getzooreport_report_nn (
  zooport_getzooreport_fkey VARCHAR (16)
    FOR BIT DATA NOT NULL
    OPTIONS(FOREIGN_KEY 'ZOOPORT_GETZOOREPORT_NN'),
  zookeeper_id VARCHAR (48) OPTIONS(XPATH './ns1:Zookeeper/@ns1:id'),
  report_name VARCHAR (48) OPTIONS(XPATH './ns1:Zookeeper/ns1:Name/text()'),
  report_numberfingersleft VARCHAR (48)
    OPTIONS(XPATH './ns1:Zookeeper/ns1:NumberFingersLeft/text()'),
  zooport_getzooreport_pkey VARCHAR (16)
    FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER "zooserver" OPTIONS(
  XPATH './ns1:Zoo' ,
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
              ns1="http://myzoo.com" ');
```

---

Abbildung 36. Zoobericht (zooreport) – untergeordnetes Element des Kurznamens 'zooport\_getzooreport\_nn'

---

```

CREATE NICKNAME zooport_getzooreport_report_report_nn (
    zooport_getzooreport_fkey VARCHAR (16) FOR BIT DATA NOT NULL
    OPTIONS(FOREIGN_KEY 'zooport_getzooreport_report_nn'),
    report_name VARCHAR (48)
    OPTIONS(XPATH './ns1:Animal/ns1:Name/text()'),
    report_species VARCHAR (48)
    OPTIONS(XPATH './ns1:Animal/ns1:Species/text()'),
    report_lot VARCHAR (48) OPTIONS(XPATH './ns1:Animal/ns1:Lot/text()'))
FOR SERVER "zooserver" OPTIONS(
    XPATH './ns1:Zookeeper/ns1:AnimalCareList' ,
    NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/" ,
    ns1="http://myzoo.com" ');

```

---

*Abbildung 37. Zoobericht (zooreport) – untergeordnetes Element von 'zooport\_getzooreport\_report\_nn'*

Das Schema enthält einige Elemente, die wiederholt werden, bzw. Folgeelemente. Diese wiederholten Elemente werden zu untergeordneten Kurznamen des übergeordneten Kurznamens, wie in Abb. 35 auf Seite 454, Abb. 36 auf Seite 454 und Abb. 37 gezeigt. So sind beispielsweise 'zooname', 'count', 'lastmodified' und 'zookeeper' Elemente von 'zoo'. Das Element 'zoo' enthält 0 oder mehr Elemente von 'zookeeper'. Der Rootkurzname 'zoo' enthält die Spalten 'zooname', 'count' und 'lastmodified'. Ein untergeordneter Kurzname ('zookeeper') wird vom Discovery-Tool der DB2-Steuerzentrale erstellt, um die sich wiederholenden Elemente von 'zookeeper' zu beschreiben. Das dritte Element ('animalcarelist') in der Spalte 'zookeeper' enthält ebenfalls 0 oder mehr Elemente und wird somit zum untergeordneten Kurznamen 'zooport\_getzooreport\_report\_report\_nn'. Die folgende Abbildung zeigt die Kurznamenhierarchie:

---

**Rootkurzname: zooport\_getzooreport\_nn**

Zoo (übergeordneter Kurzname):

- ZooName
- Count
- LastModified
- ZooKeeper (es gibt 0 oder mehr ZooKeeper-Elemente)

**Untergeordneter Kurzname: zooport\_getzooreport\_report\_nn**

ZooKeeper-Elemente

- Name
- NumberFingersLeft
- AnimalCareList (es gibt 0 oder mehr Animal-Elemente)

**Untergeordneter Kurzname: zooport\_getzooreport\_report\_report\_nn**

Animal

- Name
  - Species
  - Lot
- 

*Abbildung 38. Übergeordneter Kurzname —> untergeordneter Kurzname —> Kurznamenhierarchien*

Die nachstehende Anweisung stellt eine typische Abfrage dar, die Sie für die Kurznamen absetzen könnten, um auf den Web-Service für den Zoobericht zuzugreifen. Wenn Sie diese Anweisung absetzen, rufen Sie die Informationen aus dem Zoobericht auf Grundlage einer spezifischen ID ab sowie auf Grundlage dessen, wo die Primär- und Fremdschlüssel der Zooberichte der untergeordneten Kurznamen übereinstimmen.

```
SELECT * FROM zooport_getzooreport_nn ,
        zooport_getzooreport_report_nn zk ,
        zooport_getzooreport_report_report_nn a
WHERE zooid='1'AND zooport_getzooreport_pkey=zk.zooport_getzooreport_fkey
and zk.zooport_getzooreport_pkey=a.zooport_getzooreport_fkey;
```

### Beispiel 3: Option 'Late Binding'

Das folgende Beispiel zeigt, wie Sie die Option 'Late Binding' verwenden können. Sie können diese Option über die DB2-Steuerzentrale oder über eine DB2-Befehlszeile verwenden. Wenn Sie die Kurznamenoptionen URL und SOAPACTION definieren und die Spaltenoptionen URLCOLUMN und SOAPACTIONCOLUMN aktivieren, wenn Sie einen Kurznamen erstellen, werden die Late Binding-Funktionen verwendet. Die DB2-Steuerzentrale erstellt zwei Spaltenoptionen (URLCOLUMN und SOAPACTIONCOLUMN) und setzt die Werte der Spalten auf yes (ja).

Das nachstehende Beispiel bezieht sich auf einen Web-Service, der Preisangebote für Ersatzteile zur Verfügung stellt. Dieser Web-Service wird von allen Lieferanten eines Unternehmens implementiert. Es folgt die Anweisung CREATE NICKNAME, die die Definitionen für URLCOLUMN und SOAPACTIONCOLUMN enthält:

```
CREATE NICKNAME GetPartQuote(
  partnumber INTEGER OPTIONS (TEMPLATE'&column'),
  price FLOAT OPTIONS (XPATH './price')),
  urlcol VARCHAR(100) OPTIONS (URLCOLUMN 'Y'),
  soapactioncol VARCHAR(100) OPTIONS (SOAPACTIONCOLUMN 'Y'),
FOR SERVER myServer
OPTIONS (
  ...
  SOAPACTION 'http://example.com/GetPartPrice' ,
  URL 'http://mycompany.com:9080/GetPartPrice'',
  ...
)
```

Um Preisangebote von allen Lieferanten mit einer einzigen Abfrage abrufen zu können, benötigen Sie die Werte, die jeder Lieferant für die Spaltenoptionen SOAPACTION und URL verwendet. Die Abfrage sieht wie folgt aus:

```
SELECT * FROM supplier_endpoints p,
        GetPartQuote q
WHERE partnumber=1234 AND
      p.url=q.urlcol AND
      p.soapaction=q.soapactioncol;
```

Die lokale Tabelle 'supplier\_endpoints' enthält alle URLs und SOAP-Adressen, mit denen Sie den Web-Service aufrufen können. Sie können eine Preisklausel ORDER BY einschließen, um für das betreffende Ersatzteil den Lieferanten mit dem günstigsten Preis zu ermitteln.

#### Beispiel 4: Spaltenoption ESCAPE\_INPUT

Sie können XML-Fragmente als Eingabewerte in Ihre Abfrage einschließen. Wenn Sie einen Kurzname registrieren, schließen Sie die Spaltenoption ESCAPE\_INPUT=N ein. Mit dieser Option werden die Sonderzeichen (wie beispielsweise < und >) in XML-Fragmenten in den Eingabewerten beibehalten.

Wenn ein Schema sich wiederholende Eingabewert enthält, die es erforderlich machen würden, XML als Teil der SOAP-Nachricht zu senden, können Sie die Spaltenoption ESCAPE\_INPUT verwenden, um die Ausgabenachricht mit der korrekten XML zu erstellen.

Der Web-Service 'zoo' beispielsweise schließt eine Operation ein, um einen neuen Zoowärter (zookeeper) und die ihm zugeordneten Tiere (animals) hinzuzufügen. Im Schema für dieses Beispiel kann eine Tierpflegeliste (AnimalCareList) mehrere Tiere enthalten.

```
CREATE NICKNAME add_zookeeper(  
  zookeeper_id VARCHAR(48) OPTIONS(TEMPLATE '...'),  
  name VARCHAR(48) OPTIONS(TEMPLATE '...'),  
  numberfingersleft VARCHAR(48) OPTIONS(TEMPLATE '...'),  
  animals VARCHAR(3000) OPTIONS( TEMPLATE '...' , ESCAPE_INPUT 'N')  
  ...
```

Um einen neuen Zoowärter (zookeeper) mit zwei Tieren (animals) hinzuzufügen, setzen Sie eine Abfrage ähnlich dem folgenden Beispiel ab:

```
SELECT * FROM add_zookeeper  
  WHERE zookeeper_ID='37' AND  
        name='Amit Tsunami' AND  
        numberfingersleft='3' AND  
        animals='<AnimalCareList xmlns="http://myzoo.com">  
          <Animal>  
            <Name>Larry</Name>  
            <Species>Gorilla</Species>  
            <Lot>7</Lot>  
          </Animal>  
          <Animal>  
            <Name>Bill</Name>  
            <Species>Chimpanzee</Species>  
            <Lot>8H</Lot>  
          </Animal>  
        </AnimalCareList>';
```

Der Kurzname 'add\_zookeeper' ist eine Web-Services-Operation, die den Status des Web-Service ändern oder Informationen aktualisieren kann. Obwohl nicht relationale Wrapper nicht aktualisiert werden können, aktualisiert die Anweisung SELECT in diesem Beispiel die Zoodaten, um einen neuen Zoowärter hinzuzufügen.

Sie können die Spaltenoption ESCAPE\_INPUT auch für ein Schema einsetzen, das ein Element wie beispielsweise 'xsd:anyType' verwendet. In diesem Fall ist der Typ des Elements unbekannt. Sie können die Spaltenoption ESCAPE\_INPUT in der Eingabespalte für dieses Element verwenden, um beliebige XML-Fragmente für Ihre Eingabe anzugeben.



---

## Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services

### Gleichheitszeichen als Vergleichselement

Als Vergleichselement für Eingabespalten ist ausschließlich das Gleichheitszeichen gültig. Für Ausgabespalten sind alle Vergleichselemente gültig.

Das nachstehende Beispiel gibt eine Fehlermeldung zurück, die darüber informiert, dass das Vergleichselement für die betreffende Spalte nicht unterstützt wird. In diesem Beispiel ist die Spalte zipcode eine Eingabespalte:

```
SELECT return FROM gettemp WHERE zipcode<'95141'
```

Das nachstehende Beispiel zeigt eine gültige Abfrage, bei der ein Gleichheitszeichen als Vergleichselement für die Eingabespalte verwendet wird. Der Kurzname 'customers' wird mit einer lokalen DB2 UDB-Tabelle verknüpft, die Kunden-IDs (custid) enthält. Die Abfrage enthält ein zusätzliches Vergleichselement für die Spalte 'Sales', bei der es sich ausschließlich um eine Ausgabespalte handelt.

```
SELECT a.name, a.address
       FROM customers a, local_table b
       WHERE
           a.customer_id=b.custid AND
           a.Sales > 300000;
```

### Vergleichselemente für erforderliche Eingabespalten

Sie müssen Werte in Vergleichselementen mit Gleichheitszeichen für alle erforderlichen Eingabespalten in Ihren SQL-Abfragen für die Kurznamenhierarchie, auf die Sie verweisen, angeben. Für alle Abfragen, die gegen diese Einschränkung verstoßen, gibt der Wrapper den SQLCODE-Wert 901 zurück.

### Vergleichselemente IN oder OR

Bei WebSphere Business Integration-Wrappern und Web-Services-Wrappern sind für Eingabespalten keine IN- oder OR-Vergleichselemente zulässig.

Die nachstehenden Beispiele zeigen ungültige Abfragen. Der Kurzname 'customers' verfügt über eine erforderliche Eingabespalte namens 'customer\_id':

```
SELECT * FROM customers
       WHERE customer_id IN (12345, 67890, 11223);
SELECT * FROM customers
       WHERE customer_id IN (SELECT custid FROM local_table); )
```

Bei den WebSphere Business Integration-Wrappern können Sie jedoch IN-Vergleichselemente für Listen mit erforderlichen Eingabespalten verwenden, wenn Sie einen eindeutigen Index (UNIQUE INDEX) mit dem Parameter SPECIFICATION ONLY für die erforderlichen Eingabespalten definieren:

```
CREATE UNIQUE INDEX myuindex ON customers(customer_id) SPECIFICATION ONLY;
```



## Verknüpfungen für optionale Eingabespalten

Das nachstehende Beispiel zeigt eine Einschränkung beim Verknüpfen von optionalen Eingabespalten. Sie können keine optionalen Eingabespalten aus einer lokalen Tabelle oder einem lokalen Kurznamen verknüpfen. Wenn das WSDL-Dokument eine Eingabekurznamenspalte als optional generiert und Sie diese Spalte in einer Verknüpfung verwenden wollen, müssen Sie die DDL editieren, um die betreffende Spalte in eine erforderliche Eingabespalte zu ändern.

In diesem Beispiel wird der Kurzname `order` des Web-Services-Wrappers mit der optionalen Eingabespalte `shipping_method` erstellt. Die folgende Anweisung ist eine gültige Abfrage, da sie im Vergleichselement ein Literal verwendet:

```
SELECT * FROM order
  WHERE part="hammer" AND shipping_method="FEDEX";
```

Wenn Sie jedoch eine lokale Tabelle namens `orderparts`, die Ersatzteile (`parts`) und Versandmethoden (`shipping_method`) definiert, in die Abfrage einschließen und die Tabelle eine optionale Spalte namens `shipping_method` enthält, ist die Anweisung ungültig:

```
SELECT * FROM
  order o, orderparts op
 WHERE
  o.part="hammer" AND
  o.shipping_method=op.shipping_method
```

Bei WebSphere Business Integration-Wrappern können Vergleichselemente für optionale Eingabespalten eines Kurznamens unter Umständen im Pushdown-Modus an den WebSphere Business Integration-Adapter gesendet werden. DB2 UDB hat die Möglichkeit, diese Vergleichselemente lokal auf die Zeilen anzuwenden, die aus der Anwendungsdatenquelle abgerufen werden. Um sicherzustellen, dass Vergleichselemente für eine bestimmte Eingabespalte stets im Pushdown-Modus an den Adapter gesendet werden, müssen Sie die Eingabespalte als erforderlich deklarieren. Jede Abfrage für die Kurznamenhierarchie muss Vergleichselementwerte für die erforderlichen Eingabespalten einschließen.

Um sicherzustellen, dass gültige Ergebnisse zurückgegeben werden, muss es sich bei den verknüpften Eingabespalten um erforderliche Spalten für Web-Services-Wrapper handeln.

## Äußere Verknüpfungen

Äußere Verknüpfungen zwischen Kurznamen, bei denen der Primärschlüssel eines übergeordneten Kurznamens und der Fremdschlüssel von untergeordneten Kurznamenspalten verwendet werden, werden nicht unterstützt.

Wenn ein übergeordnetes Element in einem XML-Dokument keine untergeordneten Elemente enthält und Sie eine innere Verknüpfung zwischen dem übergeordneten Kurznamen und dem untergeordneten Kurznamen verwenden, werden für das betreffende Element keine Zeilen zurückgegeben. Sind im SAP-System beispielsweise für einen bestimmten Kunden (`customer`) keine Informationen zur Bankverbindung (`bankdetail`) vorhanden, werden für den betreffenden Kunden keine Zeilen für den Kurznamen `'sap_bapi_customer_getdetail2_sap_customerbankdetail_NN'` zurückgegeben.

Die folgenden CREATE NICKNAME-Anweisungen definieren die Spalten, die in der Beispielabfrage verwendet werden:

```
CREATE NICKNAME sap_bapi_customer_getdetail2_NN(
...
  NAME VARCHAR(35)
    OPTIONS(XPATH './ns3:sap_customeraddress/
      ns1:sap_customeraddress/ns1:NAME/text()'),
...
  NN_PKEY VARCHAR(16) OPTIONS(PRIMARY_KEY 'YES'),
  COMPANYCODE VARCHAR(4) OPTIONS(XPATH './ns3:COMPANYCODE/text()',
    TEMPLATE '<ns3:COMPANYCODE>&column</ns3:COMPANYCODE>'),
  CUSTOMERNO VARCHAR(10) OPTIONS(XPATH './ns3:CUSTOMERNO/text()',
    TEMPLATE '<ns3:CUSTOMERNO>&column</ns3:CUSTOMERNO>'),
...
  FOR SERVER sap_server
  OPTIONS(XPATH '/ns3:sap_bapi_customer_getdetail2',
    TEMPLATE '<ns3:sap_bapi_customer_getdetail2>
      &sap_bapi_customer_getdetail2_sap_customerbankdetail_NN[0,1]
      &COMPANYCODE[0,1]
      &CUSTOMERNO[1,1]
    </ns3:sap_bapi_customer_getdetail2>',
...

```

Abbildung 39. Auszug aus dem Kurznamen 'getdetail2'

```
CREATE NICKNAME sap_bapi_customer_getdetail2_sap_customerbankdetail_NN(
  CUSTOMER VARCHAR(10) OPTIONS(XPATH './ns2:CUSTOMER/text()',
    TEMPLATE '<ns2:CUSTOMER>&column</ns2:CUSTOMER>'),
  BANK_KEY VARCHAR(15) OPTIONS(XPATH './ns2:BANK_KEY/text()',
    TEMPLATE '<ns2:BANK_KEY>&column</ns2:BANK_KEY>'),
  BANK_ACCT VARCHAR(18) OPTIONS(XPATH './ns2:BANK_ACCT/text()',
    TEMPLATE '<ns2:BANK_ACCT>&column</ns2:BANK_ACCT>'),
  CTRL_KEY VARCHAR(2) OPTIONS(XPATH './ns2:CTRL_KEY/text()',
    TEMPLATE '<ns2:CTRL_KEY>&column</ns2:CTRL_KEY>'),
  BANK_REF VARCHAR(20) OPTIONS(XPATH './ns2:BANK_REF/text()',
    TEMPLATE '<ns2:BANK_REF>&column</ns2:BANK_REF>'),
  NN_FKEY VARCHAR(16) OPTIONS(FOREIGN_KEY 'SAP_BAPI_CUSTOMER_GETDETAIL2_NN'))
  FOR SERVER sap_server
  OPTIONS(XPATH '/ns3:sap_customerbankdetail/ns2:sap_customerbankdetail',
    TEMPLATE '<ns3:sap_customerbankdetail>
      <ns2:sap_customerbankdetail>
        &CUSTOMER[0,1]
        &BANK_KEY[0,1]
        &BANK_ACCT[0,1]
        &CTRL_KEY[0,1]
        &BANK_REF[0,1]
      </ns2:sap_customerbankdetail>
    </ns3:sap_customerbankdetail>',
...

```

Abbildung 40. Auszug aus dem Kurznamen 'customerbankdetail'

In folgendem Beispiel gibt die Abfrage keine Zeilen zurück, da eine Bedingung für eine innere Verknüpfung zwischen den beiden Kurznamen vorhanden ist:

```
SELECT a.name, b.bank_key
FROM sap_bapi_customer_getdetail2_NN a,
     sap_bapi_customer_getdetail2_sap_customerbankdetail_NN b
WHERE a.customerno='1234567890'
AND a.NN_PKEY=b.NN_FKEY;
```

Wenn eine Kurznamendefinition eines WebSphere Business Integration-Wrappers oder eines Web-Services-Wrappers erforderliche Eingabespalten enthält, wird keine linke äußere Verknüpfung zwischen diesem Kurznamen und einer anderen lokalen DB2 UDB-Tabelle bzw. anderen Kurznamen unterstützt.

#### **Zugehörige Konzepte:**

- „Option TEMPLATE auf Kurznamen- und Spaltenebene“ auf Seite 151

#### **Zugehörige Tasks:**

- „Hinzufügen von Geschäftsanwendungsdatenquellen zu einem System zusammengeschlossener Datenbanken“ auf Seite 144
- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken“ auf Seite 439

#### **Zugehörige Referenzen:**

- „Geschäftsanwendungsdatenquellen – Beispielabfragen“ auf Seite 175
- „Anweisung CREATE NICKNAME – Beispiele für den WebSphere Business Integration-Wrapper“ auf Seite 158
- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463

## Web-Services-Datenquellen – Beispielabfragen

### Beispiel 1: Verwenden gespeicherter Abfragetabellen

Gespeicherte Abfragetabellen werden verwendet, um die Ergebnisse einer Abfrage im lokalen Cache zwischenspeichern und um die Leistung von Abfragen zu verbessern. Zum Erstellen gespeicherter Abfragetabellen können Sie Kurznamen aus Web-Services-Datenquellen verwenden. Bei einigen Abfragen kann die Datenbank automatisch ermitteln, ob die gespeicherte Abfragetabelle eine Abfrage beantworten kann, ohne auf die Basistabelle zuzugreifen. Die folgende Prozedur zeigt, wie eine gespeicherte Abfragetabelle erstellt und aufgefüllt wird:

1. Erstellen Sie eine lokale Tabelle oder Basistabelle:

```
CREATE TABLE mystocks(ticker VARCHAR(10));
```

Mit Hilfe der lokalen Tabelle können Sie alle Werte verwalten, die Sie zwischenspeichern wollen.

2. Fügen Sie alle Werte, die zwischengespeichert werden sollen, in die Tabelle ein:

```
INSERT INTO mystocks VALUES('IBM');
INSERT INTO mystocks VALUES('MSFT');
...
```

3. Erstellen Sie einen Web-Services-Kurznamen:

```
CREATE NICKNAME stockquote_nn (
    ticker VARCHAR(40) OPTIONS (TEMPLATE '&column'),
    price VARCHAR(16) OPTIONS (XPATH './Result/text()')
)
FOR SERVER stock_server
OPTIONS (TEMPLATE '<ticker>&column</ticker>'
        XPATH './Result/text()');
```

4. Erstellen Sie eine Sicht, die aus dem Kurznamen und der lokalen Tabelle besteht:

```
CREATE VIEW stock_quote_view (ticker, price)
AS (
    SELECT nn.ticker, nn.price
    FROM stockquote_nn nn, mystocks s
    WHERE nn.ticker=s.ticker
);
```

5. Erstellen Sie eine gespeicherte Abfragetabelle:

```
CREATE TABLE stockquote_MQT (ticker, ticker2, price)
as (SELECT nn.ticker,s.ticker as ticker2, nn.price
FROM stockquote_nn nn, mystocks s
WHERE nn.ticker=s.ticker )
DATA INITIALLY DEFERRED REFRESH DEFERRED;
```

Schließen Sie alle VARCHAR-Spalten, die im Verknüpfungsprädikat (nn.ticker und s.ticker) verwendet werden, in die Ausgabeliste der gespeicherten Abfragetabelle ein, um die Chance zu maximieren, dass die gespeicherte Abfragetabelle von DB2 Universal Database verwendet wird.

Um die Aktualisierung der gespeicherten Abfragetabelle zu verzögern, geben Sie das Schlüsselwort REFRESH DEFERRED an. Gespeicherte Abfragetabellen, die mit dem Schlüsselwort REFRESH DEFERRED angegeben werden, spiegeln keine Änderungen an der zugrundeliegenden Basistabelle wider. Verwenden Sie die Klausel DATA INITIALLY DEFERRED, damit Ihre Daten nicht als Teil der Anweisung CREATE TABLE in die Tabelle eingefügt werden.

6. Setzen Sie die Anweisung REFRESH TABLE ab, um die Tabelle aufzufüllen. Die Daten in der Tabelle spiegeln das Ergebnis der Abfrage als Momentaufnahme des Zeitpunkts wider, an dem die Anweisung REFRESH TABLE abgesetzt wird. Das folgende Beispiel füllt die Tabelle 'stockquote\_MQT' auf und setzt einen Wert für das Sonderregister für den aktuellen Aktualisierungszeitpunkt (CURRENT REFRESH AGE).

```
REFRESH TABLE stockquote_MQT;  
  
SET CURRENT REFRESH AGE any;
```

Die Abfragen, die für die Daten in der gespeicherten Abfragetabelle ausgeführt werden, sind schneller als die Abfragen, die für die Daten in der Basistabelle ausgeführt werden. Wenn Sie die gespeicherte Abfragetabelle verwenden wollen, verweisen Sie auf die Sicht und nicht auf den Kurznamen:

```
SELECT * FROM stock_quote_view  
WHERE ticker='IBM';
```

Wenn Sie eine Abfrage absetzen, um einen Wert auszuwählen, der nicht zwischen gespeichert wurde, werden 0 Zeilen zurückgegeben.

### **Beispiel 2: Absetzen von Verknüpfungen unter Verwendung der Primär- und Fremdschlüssel**

Die Spalten PRIMARY\_KEY (Primärschlüssel) und FOREIGN\_KEY (Fremdschlüssel) werden verwendet, um die Abhängigkeiten zwischen den übergeordneten und untergeordneten Kurznamen zu definieren. Jeder übergeordnete Kurzname muss über eine Primärschlüsselspaltenoption verfügen. Untergeordnete Kurznamen eines übergeordneten Kurznamens werden mit Hilfe der Fremdschlüsselspaltenoption definiert, die auf die Primärschlüsselspalte eines übergeordneten Kurznamens verweist. Ein Kurzname kann über mehrere untergeordnete Elemente verfügen; er kann jedoch nur ein übergeordnetes Element haben.

Da diese Spalten lediglich binäre Daten enthalten, werden sie mit den Schlüsselwörtern FOR BIT DATA NOT NULL definiert. Diese Definition wird beim Erstellen des Kurznamens von der DB2-Steuerzentrale generiert. Wenn Sie den Kurznamen erstellen, können Sie die Spalten PRIMARY\_KEY und FOREIGN\_KEY explizit mit FOR BIT DATA NOT NULL definieren.

Das folgende Beispiel zeigt, wie der Web-Services-Wrapper die Spalten PRIMARY\_KEY und FOREIGN\_KEY verwendet, um übergeordnete und untergeordnete Kurznamen zuzuordnen:

```

CREATE NICKNAME zooport_getzooreport_nn (
  zooid VARCHAR (48) OPTIONS(TEMPLATE '&column'),
  zoo_id VARCHAR (48) OPTIONS(XPATH './ns1:Zoo/@id'),
  report_zoo_zooname VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:ZooName/text()'),
  report_zoo_count VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:Count/text()'),
  report_zoo_lastmodified VARCHAR (48)
    OPTIONS(XPATH './ns1:Zoo/ns1:LastModified/text()'),
  nn_pk VARCHAR (16) NOT NULL OPTIONS(PRIMARY_KEY 'YES'),
  url VARCHAR (256) OPTIONS(URLCOLUMN 'Y'),
  soapaction VARCHAR (256) OPTIONS(SOAPACTIONCOLUMN 'Y')
) FOR SERVER "mytestsrvr"
OPTIONS(
  URL 'http://localhost:9080/MaelstromTest/services/ZooPort',
  SOAPACTION 'http://myzoo.com/getZooReport' ,
  TEMPLATE '<soapenv:Envelope>
    <soapenv:Body>
      <zooId>&zooId[1,1]</zooId>
    </soapenv:Body>
  </soapenv:Envelope>',
  XPATH '/soapenv:Envelope/soapenv:Body' ,
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/",
              ns1="http://myzoo.com" ');

CREATE NICKNAME zooport_getzooreport_report_zookeeper_nn (
  nn_fk VARCHAR (16) NOT NULL
    OPTIONS(FOREIGN_KEY 'ZOOPORT_GETZOOREPORT_NN'),
  zookeeper_id VARCHAR (48) OPTIONS(XPATH './@id'),
  report_zookeeper_name VARCHAR (48) OPTIONS(XPATH './ns1:Name/text()'),
  zookeeper_numberfingersleft VARCHAR(48)
    OPTIONS(XPATH './ns1:NumberFingersLeft/text()'),
  nn_pk VARCHAR (16) NOT NULL OPTIONS(PRIMARY_KEY 'YES')
)
FOR SERVER "MYTESTSRVR" OPTIONS(
  XPATH './ns1:Zoo/ns1:Zookeeper' ,
  NAMESPACES ' soapenv="http://schemas.xmlsoap.org/soap/envelope/",
              ns1="http://myzoo.com" ');

```

Der Fremdschlüssel 'nn\_fk' im Kurznamen 'zooport\_getzooreport\_report\_zookeeper\_nn' verweist auf den übergeordneten Kurznamen 'zooport\_getzooreport\_nn' in der Fremdschlüsselloption. Die ausgewiesenen Primär- und Fremdschlüsselspalten für Kurznamen entsprechen nicht den Daten Ihres WSDL-Dokuments, da diese Kurznamenspalten Schlüssel enthalten, die vom Wrapper generiert werden. Diese Schlüssel identifizieren eine Abhängigkeit zwischen den übergeordneten und untergeordneten Kurznamen, die nur innerhalb einer Abfrage eindeutig ist. Enthält der untergeordnete Kurzname eine Eingabespalte, verweist die übergeordnete Kurznamenoptionsschablone auf diesen untergeordneten Kurznamen in der Struktur der Kurznamenoptionsschablone.

Die folgende SQL-Anweisung verknüpft die übergeordneten und untergeordneten Kurznamen:

```
SELECT *
FROM   zooport_getzooreport_nn a,
       zooport_getzooreport_report_zookeeper_nn z,

WHERE  a.nn_pk = z.nn_fk
       AND a.zooid = 100
       ;
```

Die nachstehende Beschreibung erläutert, wie der Web-Services-Wrapper die Kurznamen- und Spaltenoptionen TEMPLATE und XPATH während der Ausführung von Abfragen verwendet. Die Beschreibung ist nicht als Beispiel einer spezifischen Implementierung gedacht.

Wenn Sie die Primär- und Fremdschlüsselspalten verknüpfen, sendet der Web-Services-Wrapper eine Nachricht an den Web-Services-Provider. Daraufhin gibt der Web-Services-Provider eine Zeilengruppe zurück. Der Wrapper generiert eine Nachricht für den übergeordneten Kurznamen. Hierbei werden die Werte der Eingabespalte (a.zooid = 100) aus der Abfrage für den Verweis in der Spaltenoptionsschablone (ZOOID VARCHAR (48) OPTIONS(TEMPLATE '&column')) und anschließend alle Spaltenverweise in der Kurznamenschablonenoption (<zooId>&zooId[1,1]</zooId>) ersetzt. Die Kurznamenschablonenoption kann Spaltenverweise oder Verweise auf untergeordnete Kurznamen enthalten. Anschließend wird die Nachricht an den Web-Service gesendet.

Der Wrapper generiert die Zeilen für einen Kurznamen, indem die Kurznamenoption XPATH auf das vom Web-Service zurückgegebene Dokument angewendet wird. Gibt die Kurznamenoption XPATH mehrere XML-Fragmente zurück, enthält der Kurzname mehrere Spalten. Die Spaltenoption XPATH wird auf die resultierenden XML-Fragmente angewendet, die die Zeilen zum Abrufen der Spaltenwerte darstellen. Verfügt ein Kurzname über mindestens ein indirektes übergeordnetes Element, werden alle XPATH-Ausdrücke übergeordneter Kurznamen von oben nach unten in der Hierarchie angewendet, bevor die Kurznamenoption XPATH und die Spaltenoption XPATH für den betreffenden Kurznamen angewendet werden.

#### **Zugehörige Tasks:**

- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 439
- „Registrieren des Web-Services-Wrappers“ auf Seite 440
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442

#### **Zugehörige Referenzen:**

- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Nachrichten für den Web-Services-Wrapper“ auf Seite 468
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172

---

## Web-Services-Wrapper - Sicherheit

Der Web-Services-Wrapper unterstützt HTTPS als Übertragungsprotokoll für SOAP-Nachrichten. HTTPS ist ein standardmäßiges Verschlüsselungsprotokoll, das von vielen Providern von Web-Services verwendet wird. Das von den Web-Services-Providern generierte WSDL-Dokument enthält 'https://' in der URL. Die SOAP-Nachrichten in der HTTP-Anforderung oder der HTTP-Antwort werden verschlüsselt.

Wenn der Web-Service HTTPS als Übertragungsprotokoll verwendet, werden die SSL-Zertifikate, die der Server zur Identifikation sendet, nicht vom Web-Services-Wrapper geprüft. Der Web-Services-Wrapper kann Web-Services mit selbst unterzeichneten Zertifikaten aufrufen.

Der Web-Services-Wrapper verwendet die Anweisung CREATE USER MAPPING zur Unterstützung der HTTP-Authentifizierung. Mit der Anweisung CREATE USER MAPPING können Sie die Benutzer-ID eines Servers mit zusammengesetzten Datenbanken zu einer Benutzer-ID und einem Kennwort eines Web-Service zuordnen. Die Entwickler von Wrappern liefern eine Benutzer-ID und ein Kennwort mit einer Benutzerzuordnungsanweisung wie beispielsweise der folgenden:

```
CREATE USER MAPPING
  FOR RSPALTEN SERVER S1
  OPTIONS ( REMOTE_AUTHID 'SYSTEM', REMOTE_PASSWORD 'MANAGER' )
```

Wenn auf einen Web-Services-Kurznamen auf dem Server S1 zugegriffen wird, wird die HTTP-Anforderung mit SYSTEM als Benutzer-ID und MANAGER als Kennwort gesendet. Die Benutzerzuordnung ist optional. Wenn Sie keine Benutzerzuordnung angeben, wird möglicherweise eine Fehlermeldung angezeigt, wenn der Web-Services-Provider Authentifizierungsinformationen erwartet. Einige Server schränken den Zugriff auf einen Service möglicherweise durch Authentifizierung ein. Die Informationen im WSDL-Dokument geben keine Auskunft darüber, ob Authentifizierung erforderlich ist oder nicht.

### Zugehörige Konzepte:

- „Der Web-Services-Wrapper und das WSDL-Dokument“ auf Seite 433

### Zugehörige Tasks:

- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 439
- „Registrieren des Web-Services-Wrappers“ auf Seite 440
- „Registrieren von Kurznamen für Web-Services-Datenquellen“ auf Seite 442

### Zugehörige Referenzen:

- „Nachrichten für den Web-Services-Wrapper“ auf Seite 468



## Nachrichten für den Web-Services-Wrapper

Die folgende Tabelle erläutert einige der häufigsten Fehlernachrichten, die bei Verwendung des Web-Services-Wrappers angezeigt werden können.

Tabelle 90. Häufige Fehlernachrichten

Fehler	Beschreibung	Benutzeraktion
SQL1822N Es wurde ein unerwarteter Fehlercode "SOAP-Fehler" von der Datenquelle "wswrap.svl.ibm". Zugeordneter Text und Token sind "java.lang.Exception: HTTP-URIs". SQLSTATE=560BD	Der Web-Services-Provider gab einen SOAP-Fehler zurück.	<ul style="list-style-type: none"> <li>Die Eingabe ist möglicherweise nicht korrekt. Überprüfen Sie, ob Ihre Eingabeargumente korrekt sind.</li> <li>Möglicherweise ist beim Web-Services-Provider ein Problem aufgetreten. Setzen Sie sich mit dem Eigner des Web-Service in Verbindung.</li> </ul>
SQL30081N Kommunikationsfehler. Verwendetes Kommunikationsprotokoll: "SOAP". Verwendete Kommunikations-API: "HTTP". Position, an der der Fehler festgestellt wurde: "localhos". Kommunikationsfunktion, die den Fehler feststellte: "connect". Protokollspezifische(r) Fehlercode(s): "38309", "10061", "1". SQLSTATE=08001	Der Web-Services-Provider war am Anschluss (Port) oder an der URL nicht empfangsbereit.	<ul style="list-style-type: none"> <li>Überprüfen Sie, ob alle verwendeten URLs auch gültig sind.</li> <li>Überprüfen Sie, ob die Anschlussinformationen korrekt sind.</li> <li>Stellen Sie sicher, dass der Server aktiv ist.</li> </ul>
SQL30081N Kommunikationsfehler. Verwendetes Kommunikationsprotokoll: "SOAP". Verwendete Kommunikations-API: "HTTP". Position, an der der Fehler festgestellt wurde: "doesntexist.ibm". Kommunikationsfunktion, die den Fehler feststellte: "*". Protokollspezifische(r) Fehlercode(s): "38308", "*", "0". SQLSTATE=08001	Der Hostname des Web-Services-Providers befindet sich nicht auf dem Domänennamensserver.	Stellen Sie sicher, dass sich der Name Ihres Hosts auf dem Namensserver befindet.
SQL30081N Kommunikationsfehler. Verwendetes Kommunikationsprotokoll: "SOAP". Verwendete Kommunikations-API: "HTTP". Position, an der der Fehler festgestellt wurde: "www.ibm.com". Kommunikationsfunktion, die den Fehler feststellte: "*". Protokollspezifische(r) Fehlercode(s): "38312", "*", "0". SQLSTATE=08001	HTTP-Rückkehrcode, ähnlich dem von einem Browser häufig gemeldeten Fehler 404.	Ermitteln Sie, ob der Server Fehler an Ihre Anwendung zurückgibt. Führen Sie einen DB2 UDB-Trace durch, um zu ermitteln, welche Antwort der Server zurückgibt.
SQL30081N Kommunikationsfehler. Verwendetes Kommunikationsprotokoll: "SOAP". Verwendete Kommunikations-API: "HTTP". Position, an der der Fehler festgestellt wurde: "". Kommunikationsfunktion, die den Fehler feststellte: "*". Protokollspezifische(r) Fehlercode(s): "38304", "*", "0". SQLSTATE=08001	Die URL ist nicht korrekt. Der SQLSTATE-Wert 38304 weist möglicherweise darauf hin, dass das Protokoll unbekannt ist. Der SQLSTATE-Wert 38305 gibt einen Syntaxfehler in der URL an.	Prüfen Sie, ob Ihr WSDL-Dokument eine gültige URL-Syntax und ein gültiges Protokoll enthält.

Tabelle 90. Häufige Fehlermeldungen (Forts.)

Fehler	Beschreibung	Benutzeraktion
SQL1822N Es wurde ein unerwarteter Fehlercode "SAXException" von der Datenquelle "wswrap.svl.ibm" empfangen. Zugeordneter Text und Token sind "Erwartetes Ende der Markierung 'ns0:Mi". SQLSTATE=560BD	Die Antwort enthält einen Syntaxanalysefehler in der XML-Ausgabe.	Prüfen Sie, ob die vom Server zurückgegebene XML-Ausgabe korrekt ist. Führen Sie einen DB2 UDB-Trace aus, um zu ermitteln, welche Antwort der Web-Services-Provider zurückgibt. Sie können den Web-Service auch mit einem anderen Tool aufrufen, um sicherzustellen, dass die Antwort des Web-Service gültig ist.

**Zugehörige Konzepte:**

- „Der Web-Services-Wrapper und das WSDL-Dokument“ auf Seite 433

**Zugehörige Tasks:**

- „Hinzufügen von Web-Services-Datenquellen zu einem Server mit zusammengesetzten Datenbanken“ auf Seite 439
- „Registrieren des Web-Services-Wrappers“ auf Seite 440

**Zugehörige Referenzen:**

- „Anweisung CREATE NICKNAME – Beispiele für den Web-Services-Wrapper“ auf Seite 451
- „Web-Services-Datenquellen – Beispielabfragen“ auf Seite 463
- „Einschränkungen bei Abfragen für Wrapper für Geschäftsanwendungen und Web-Services“ auf Seite 172



---

## Kapitel 24. Konfigurieren des Zugriffs auf XML-Datenquellen

In diesem Kapitel wird erklärt, wie Sie Ihren Server mit zusammengeschlossenen Datenbanken für den Zugriff auf Daten konfigurieren können, die in XML-Datenquellen gespeichert sind. Der Zugriff auf XML-Datenquellen kann über die DB2-Steuerzentrale oder durch Absetzen von SQL-Anweisungen konfiguriert werden.

Dieses Kapitel enthält folgende Informationen:

- Erläuterungen zu XML.
- Liste der auszuführenden Tasks.
- Beispiele zu den benötigten SQL-Anweisungen.
- Liste der dem XML-Wrapper zugeordneten Fehlernachrichten.

---

### Was ist XML?

XML (Extensible Markup Language) ist ein universelles Format für strukturierte Dokumente und Daten. XML-Dateien haben die Dateierweiterung `xml`. Ebenso wie HTML verwendet auch XML Befehle (in die Zeichen `>` und `<` eingeschlossene Wörter), um Daten in einem Dokument zu strukturieren. Abb. 41 zeigt ein XML-Dokument als Beispiel.

---

```
<?xml version="1.0" encoding=UTF-8"?>
<doc>
  <customer id='123'>
    <name>...</name>
    <address>...</address>
    ...
    <order>
      <amount>...</amount>
      <date>...</date>
      <item quant='12'>
        <name>...</name>
      </item>
      <item quant='4'>...</item>
      ...
    </order>
    <order>...</order>
    ...
    <payment>
      <number>...</number>
      <date>...</date>
    </payment>
    <payment>>...</payment>
    ...
  </customer>
  <customer id='124'>...</customer>
</doc>
```

---

Abbildung 41. XML-Beispieldokument

## Funktionsweise des XML-Wrappers

Der XML-Wrapper ermöglicht die Verwendung von SQL zum Abfragen folgender Datentypen:

- Externe XML-Dokumente, die in einer einzelnen Datei gespeichert sind
- Mehrere XML-Dateien in einem Verzeichnispfad
- Ferne XML-Dateien, auf die mit einem Uniform Reference Identifier (URI) verwiesen wird
- In relationalen Spalten gespeicherte XML-Dokumente

Abb. 42 zeigt, wie der XML-Wrapper mit Ihrem System zusammengeschlossener Datenbanken arbeitet.

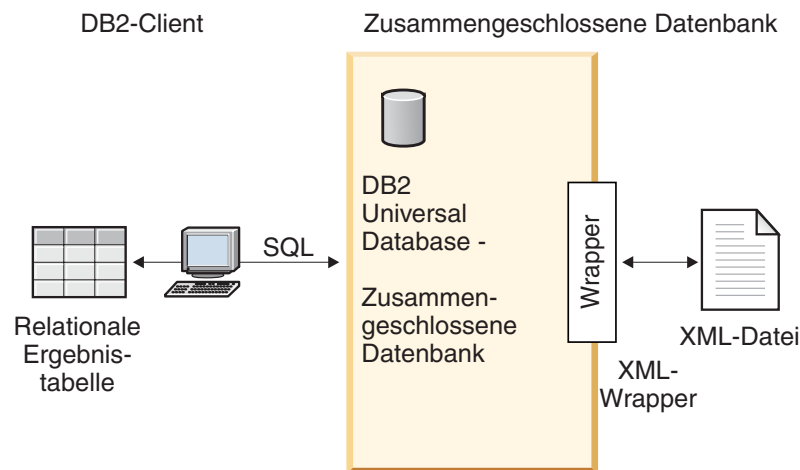


Abbildung 42. Funktionsweise des XML-Wrappers

Der XML-Wrapper ermöglicht das Zuordnen von XML-Daten aus einer externen Datenquelle in ein relationales Schema, das aus einer Gruppe von Kurznamen besteht. Die Struktur des XML-Dokuments ist mit der eines relationalen Schemas logisch äquivalent, in dem verschachtelte und sich wiederholende Elemente als getrennte Tabellen mit Fremdschlüsseln modelliert sind.

Die einem XML-Dokument entsprechenden Kurznamen werden in einer Baumstruktur angeordnet, in der die untergeordneten Kurznamen (Kinder) Elementen zugeordnet werden, die in dem Element, das dem übergeordneten Kurznamen (Elter) entspricht, verschachtelt sind.

Werden verschachtelte Elemente wiederholt, oder haben sie Identitäten mit komplexen Strukturen, können Sie die verschachtelten Elemente jeweils mit separaten Kurznamen versehen.

Unter- und übergeordnete Kurznamen werden durch Primär- und Fremdschlüssel miteinander verbunden, die vom Wrapper generiert werden.

XPath-Ausdrücke werden verwendet, um ein XML-Dokument einem relationalen Schema zuzuordnen, das aus einer Gruppe von Kurznamen besteht. XPath ist ein Adressierungsmechanismus zum Identifizieren von Teilen einer XML-Datei, beispielsweise der Gruppen von Knoten und Attributen innerhalb einer XML-Dokumentbaumstruktur. Die XPath-Basissyntax entspricht der Dateisystemadressierung.

Jeder Kurzname wird durch einen XPath-Ausdruck definiert, der die XML-Elemente zur Darstellung einzelner Tupel identifiziert, sowie durch eine Gruppe von XPath-Ausdrücken, die angeben, wie die Spaltenwerte aus den einzelnen Elementen extrahiert werden sollen.

### Beispiel für XML-Dokumentzuordnung:

Das folgende Beispiel zeigt, wie das XML-Beispieldokument aus Abb. 41 auf Seite 471 einer Gruppe von Kurznamen zugeordnet wird, wie Eltern/Kind-Beziehungen mit Hilfe von Primär- und Fremdschlüsseln erstellt werden, wie einzelne Tupel und Spalten innerhalb der jeweiligen Elemente des Dokuments anhand von XPath-Ausdrücken definiert werden und wie eine Abfrage für das XML-Dokument ausgeführt werden kann, nachdem das Dokument für das System zusammenschlossener Datenbanken registriert worden ist.

Das XML-Beispieldokument enthält eine Gruppe von customer-Elementen (Kunden). Jedes Element schließt mehrere order- und payment-Elemente ein (Bestellung und Zahlung).

Die order-Elemente wiederum umfassen mehrere item-Elemente (Artikel).

Die Beziehung zwischen den Elementen geht aus Abb. 43 hervor.

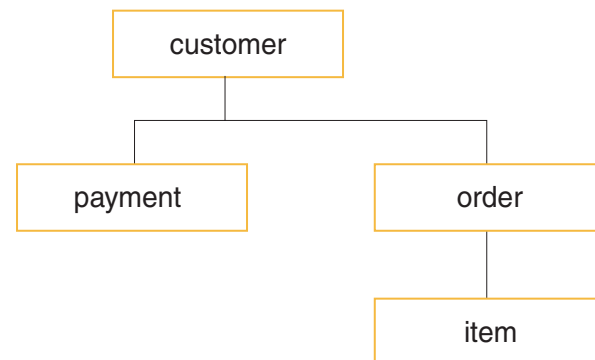


Abbildung 43. Baumstruktur des XML-Beispieldokuments

Ausgehend von dieser Struktur können Sie das XML-Dokument mit einer Anweisung CREATE NICKNAME einem relationalen Schema zuordnen, das vier Kurznamen enthält:

- customers (Kunden)
- orders (Bestellungen)
- payments (Zahlungen)
- items (Artikel)

Die Beziehungen zwischen den Kurznamen werden definiert, indem jeder Kurzname als übergeordneter Kurzname (Elter) bzw. untergeordneter Kurzname (Kind) angegeben wird. Hierzu werden Kurznamenspaltenoptionen mit spezifischen Primär- bzw. Fremdschlüsseln verwendet. Jedem übergeordneten Kurznamen muss eine spezifische Spalte mit einer Primärschlüsselspaltenoption zugeordnet werden. Untergeordnete Kurznamen eines übergeordneten Kurznamens werden mit Hilfe der Fremdschlüsselspaltenoption definiert, die auf die Primärschlüsselspalte eines übergeordneten Kurznamens verweist. Die zugeordneten Primär- und Fremdspalten für Kurznamen entsprechen nicht den Daten Ihres XML-Dokuments, da diese Kurznamenspalten Schlüssel enthalten, die vom Wrapper generiert werden.

Ein Kurzname kann über mehrere untergeordnete Kurznamen verfügen; er kann jedoch nur einen übergeordneten Kurznamen haben. Der Rootkurzname hat keinen übergeordneten Kurznamen.

Im XML-Beispieldokument hat der Kurzname 'customer' einen definierten Primärschlüssel, und die Kurznamen 'order', 'payment' und 'item' haben definierte Fremdschlüssel, die auf den übergeordneten Kurznamen verweisen. Die Fremdschlüssel für die Kurznamen 'order' und 'payment' verweisen auf den Kurznamen 'customer', und der Fremdschlüssel des Kurznamens 'item' verweist auf den Kurznamen 'order'.

Um die XML-Elemente zu identifizieren, die einzelne Tupel darstellen, wird ein XPath-Ausdruck erstellt. In diesem Beispiel kann mit Hilfe des XPath-Ausdrucks '/doc/customer' auf alle customer-Elemente und mit Hilfe des XPath-Ausdrucks './order' auf alle order-Elemente verwiesen werden. Der Punkt im XPath-Ausdruck './order' gibt an, dass die Tupel des jeweiligen order-Elements innerhalb der Tupel des entsprechenden customer-Elements verschachtelt sind.

Eine Gruppe von XPath-Ausdrücken wird erstellt, um anzugeben, wie die Spaltenwerte aus den einzelnen Elementen extrahiert werden sollen. In diesem Beispiel kann mit Hilfe des XPath-Ausdrucks id auf das Attribut './@id' der customer-Elemente (jetzt eine im Kurznamen definierte Spalte) verwiesen werden. Mit Hilfe des XPath-Ausdrucks './name' kann auf das Element 'name' der customer-Elemente und mit Hilfe des XPath-Ausdrucks './address/@street' auf das Element 'address' der customer-Elemente verwiesen werden.

Nachdem das XML-Dokument mit Hilfe der Anweisung CREATE NICKNAME einer Gruppe von Kurznamen zugeordnet wurde, erfolgt die Definition jedes Kurznamens anhand von Primär- und Fremdschlüsseln als übergeordneter oder untergeordneter Kurzname (Elter oder Kind). Dabei werden XPath-Ausdrücke verwendet, die einzelne Tupel und Spalten innerhalb aller Elemente des Dokuments definieren. Danach können SQL-Abfragen für das XML-Dokument ausgeführt werden.

#### **Zugehörige Konzepte:**

- „Datenzuordnungen zwischen Kurznamen und XML-Dokumenten“ auf Seite 479

#### **Zugehörige Tasks:**

- „Hinzufügen von XML zu einem System zusammengesetzter Datenbanken“ auf Seite 474

---

## **Hinzufügen von XML zu einem System zusammengesetzter Datenbanken**

### **Hinzufügen von XML zu einem System zusammengesetzter Datenbanken**

Um den Server mit zusammengesetzten Datenbanken für den Zugriff auf XML-Datenquellen zu konfigurieren, müssen Sie für den Server Informationen zu den Datenquellen und Objekten bereitstellen, auf die Sie zugreifen möchten.

Sie können den Zugriff für den Server mit zusammengesetzten Datenbanken auf XML-Datenquellen mit Hilfe der DB2-Steuerzentrale oder über die DB2-

Befehlszeile konfigurieren. Die DB2-Steuerzentrale beinhaltet einen Assistenten, der Sie durch die erforderlichen Schritte für die Konfiguration des Servers mit zusammengeschlossenen Datenbanken führt.

**Voraussetzungen:**

- DB2 Information Integrator muss auf dem Server installiert sein, der als Server mit zusammengeschlossenen Datenbanken fungiert.
- Eine zusammengeschlossene Datenbank muss sich auf dem Server mit zusammengeschlossenen Datenbanken befinden.

**Vorgehensweise:**

Um eine XML-Datenquelle zu einem Server zusammengeschlossener Datenbanken hinzuzufügen, gehen Sie wie folgt vor:

1. Registrieren Sie den XML-Wrapper.
2. Registrieren Sie die XML-Serverdefinition.
3. Registrieren Sie Kurznamen für die XML-Datenquellen.
4. Erstellen Sie Sichten für zusammengeschlossene Datenbanken für Nicht-Rootkurznamen.

Ein Rootkurzname ist ein Kurzname auf der höchsten Ebene der Kurznamenhierarchie. Ein Nicht-Rootkurzname ist ein Kurzname, der einen übergeordneten Kurznamen in der Kurznamenhierarchie hat. Ein XML-Dokument kann Rootkurznamen enthalten, die kein Element der höchsten Ebene darstellen.

Sie können die Anweisungen von der DB2-Steuerzentrale oder von einem DB2-Befehlszeilenprozessor aus ausführen. Nachdem Sie den XML-Wrapper dem System zusammengeschlossener Datenbanken hinzugefügt haben, können Sie Abfragen für XML-Datenquellen ausführen.

**Zugehörige Konzepte:**

- „Installationsprozess von DB2 Information Integrator - Übersicht“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

**Zugehörige Tasks:**

- „Erstellen einer zusammengeschlossenen Datenbank“ auf Seite 58
- „Registrieren des XML-Wrappers“ auf Seite 475
- „Registrieren des Servers für XML-Datenquellen“ auf Seite 477
- „Registrieren von Kurznamen für XML-Datenquellen“ auf Seite 482
- „Erstellen von Sichten für zusammengeschlossene Datenbanken für Nicht-Stammkurznamen (XML-Wrapper)“ auf Seite 487

**Zugehörige Referenzen:**

- „Unterstützte Betriebssysteme für DB2 Information Integrator (32-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Unterstützte Betriebssysteme für DB2 Information Integrator (64-Bit)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

## Registrieren des XML-Wrappers

Die Registrierung des XML-Wrappers ist ein Teil der umfangreicheren Task des Hinzufügens von XML-Datenquellen zu einem Server mit zusammengeschlossenen Datenbanken.



Sie müssen einen Wrapper für den Zugriff auf XML-Datenquellen registrieren. Wrapper werden von Servern mit zusammengeschlossenen Datenbanken für die Kommunikation und den Abruf von Daten aus Datenquellen verwendet. Wrapper werden als Gruppe von Bibliotheksdateien implementiert.

**Vorgehensweise:**

Um einen Wrapper zu registrieren, geben Sie die Anweisung CREATE WRAPPER mit dem Namen des Wrappers und dem Namen der Bibliotheksdatei des Wrappers aus.

Um beispielsweise einen Wrapper mit dem Namen xml\_wrapper auf dem Server mit zusammengeschlossenen Datenbanken unter AIX zu registrieren, geben Sie folgende Anweisung aus:

```
CREATE WRAPPER xml_wrapper LIBRARY 'libdb2lxml.a';
```

Der Name der angegebenen Bibliotheksdatei des Wrappers ist vom Betriebssystem des Servers mit zusammengeschlossenen Datenbanken abhängig. Informationen zur Angabe des korrekten Namens in der Anweisung CREATE WRAPPER finden Sie in der Liste der Bibliotheksdateien von XML-Wrappern.

Die nächste Task in dieser Tasksequenz ist das Registrieren der Serverdefinition für den XML-Wrapper.

**Zugehörige Referenzen:**

- „XML-Wrapper-Bibliotheksdateien“ auf Seite 476
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

## XML-Wrapper-Bibliotheksdateien

Die folgende Tabelle enthält eine Liste der Verzeichnispfade und Bibliotheksdateinamen für den XML-Wrapper.

Wenn Sie DB2 Information Integrator installieren, werden dem in der Tabelle aufgeführten Verzeichnispfad 3 Bibliotheksdateien hinzugefügt. Wird der Server mit zusammengeschlossenen Datenbanken beispielsweise unter AIX ausgeführt, werden die Wrapperbibliotheksdateien libdb2lxml.a, libdb2lxmlf.a und libdb2lxmlu.a dem Verzeichnispfad hinzugefügt.

Wenn Sie einen Wrapper registrieren, geben Sie nur den Bibliotheksdateinamen an, der in der Tabelle aufgeführt ist.

*Tabelle 91. Speicherpositionen und Dateinamen der XML-Wrapperbibliothek*

Betriebssystem	Verzeichnispfad	Wrapperbibliotheksdatei
AIX	/usr/opt/db2_08_01/lib/	libdb2lxml.a
HP-UX	/opt/IBM/db2/V8.1/lib	libdb2lxml.sl
Linux	/opt/IBM/db2/V8.1/lib	libdb2lxml.so
Solaris	/opt/IBM/db2/V8.1/lib	libdb2lxml.so
Windows	%DB2PATH%\bin	db2lxml.dll

| %DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben  
| wird, in dem DB2 Information Integrator unter Windows installiert ist. Der  
| Standardverzeichnispfad unter Windows lautet C:\Program Files\IBM\SQLLIB.

| **Zugehörige Tasks:**

- | • „Registrieren des XML-Wrappers“ auf Seite 475

## Registrieren des Servers für XML-Datenquellen

Die Registrierung des Servers für eine XML-Datenquelle ist Teil der umfangreichen Task des Hinzufügens von XML zu einem System zusammenschlossener Datenbanken. Nach der Registrierung des Wrappers muss ein entsprechender Server registriert werden.

**Einschränkungen:**

Der XML-Wrapper verwendet nicht die Schlüsselwörter TYPE und VERSION. Werden diese Schlüsselwörter in der Anweisung CREATE SERVER verwendet, tritt ein Fehler auf. Der XML-Wrapper unterstützt keine Durchgriffssitzungen für Systeme zusammenschlossener Datenbanken.

**Vorgehensweise:**

Um den XML-Server für das System zusammenschlossener Datenbanken zu registrieren, geben Sie die Anweisung CREATE SERVER aus.

Beispiel:

```
CREATE SERVER xml_server WRAPPER my_xml;
```

Die nächste Task in dieser Tasksequenz ist das Registrieren der Kurznamen für XML-Datenquellen.

### Zugreifen auf XML-Dokumente über einen Proxy-Server

Um auf XML-Dokumente über einen Proxy-Server zuzugreifen, müssen Sie beim Erstellen der Serverdefinition Optionen angeben. Diese von Ihnen angegebenen Optionen sind abhängig von dem Typ des Proxy-Servers, auf den Sie zugreifen möchten.

Sie müssen in der Anweisung CREATE SERVER die Optionen für Proxy verwenden, wenn alle folgenden Bedingungen erfüllt sind:

- Sie möchten Daten über einen URI abrufen
- Der verwendete URI ruft Daten hinter einer Firewall (über einen Proxy) ab
- Die verwendete Firewall bzw. der verwendete Proxy ist HTTP, SOCKS4 oder SOCKS5

Fragen Sie Ihren Netzadministrator nach Informationen über den Typ des von Ihnen verwendeten Proxys und lassen Sie sich Auskunft darüber geben, welche Einstellungen Sie in den Optionen für Proxys angeben sollen.

**Beispiel für das Registrieren einer Serverdefinition für einen HTTP-Proxy-Server:**

Verwenden Sie folgende Anweisung, um eine Serverdefinition zu registrieren und einen HTTP-Proxy-Server anzugeben:

```

CREATE SERVER xml_server_h
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'HTTP', PROXY_SERVER_NAME 'proxy_h',
          PROXY_SERVER_PORT '8080');

```

### Beispiel für das Registrieren einer Serverdefinition für einen SOCKS4-Proxy-Server:

Verwenden Sie folgende Anweisung, um eine Serverdefinition zu registrieren und einen SOCKS4-Proxy-Server anzugeben:

```

CREATE SERVER xml_server_s4
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS4', PROXY_SERVER_NAME 'proxy_4',
          PROXY_SERVER_PORT '1080');

```

### Beispiel für das Registrieren einer Serverdefinition für einen SOCKS5-Proxy-Server ohne Authentifizierungsinformationen:

Verwenden Sie folgende Anweisung, um eine Serverdefinition zu registrieren und einen SOCKS5-Proxy-Server ohne Authentifizierungsinformationen anzugeben:

```

CREATE SERVER xml_server_s5
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS5', PROXY_SERVER_NAME 'proxy_5',
          PROXY_SERVER_PORT '1081');

```

### Beispiel für das Registrieren einer Serverdefinition für einen SOCKS5-Proxy-Server mit Authentifizierungsinformationen:

Verwenden Sie folgende Anweisung, um eine Serverdefinition zu registrieren und einen SOCKS5-Proxy-Server mit Authentifizierungsinformationen anzugeben:

```

CREATE SERVER xml_server_s5a
  WRAPPER xml_wrapper
  OPTIONS (PROXY_TYPE 'SOCKS5', PROXY_SERVER_NAME 'proxy_5',
          PROXY_SERVER_PORT '1081', PROXY_AUTHID 'Martin',
          PROXY_PASSWORD 'aaa', );

```

Die XML-Prüffunktion weist unter Umständen einige Einschränkungen auf, wenn sie zusammen mit der Proxyfunktion verwendet wird. Diese Einschränkungen machen sich unter folgenden Bedingungen bemerkbar:

- Sie verwenden die Proxyfunktion auf Serverebene und haben die verschiedenen Proxyoptionen eingestellt.
- Das XML-Dokument des Exemplars enthält einen Verweis auf ein externes XML-Schema, das sich außerhalb der Firewall befindet.

Liegt eine dieser beiden Bedingungen vor, versuchen Sie die Position Ihres XML-Schemas in eine Position innerhalb der Firewall zu ändern. Wenn Sie die Position des XML-Schemas ändern, müssen Sie das XML-Dokument des Exemplars hinsichtlich der neuen Position des XML-Schemas aktualisieren.

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für XML-Datenquellen“ auf Seite 482

#### Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

# Registrieren von Kurznamen für XML-Datenquellen

## Datenzuordnungen zwischen Kurznamen und XML-Dokumenten

Kurznamen entsprechen der Baumstruktur der Daten Ihres XML-Dokuments. Übergeordnete und untergeordnete Kurznamen (Eltern und Kinder) entsprechen der Stammstruktur (Root) und den verschachtelten Elementen der Datenbaumstruktur. Diese übergeordneten und untergeordneten Kurznamen werden durch Primär- und Fremdschlüssel miteinander verbunden, die in der Anweisung CREATE NICKNAME angegeben werden.

Jeder Kurzname wird durch XPath-Ausdrücke definiert, die folgende Funktionen ausführen:

- Identifizieren der XML-Elemente, die einzelne Tupel darstellen;
- Angeben, wie die Spaltenwerte aus den einzelnen Elementen extrahiert werden sollen.

Der XML-Wrapper verwendet XPath-Ausdrücke, um eine Korrespondenz zwischen den Daten im XML-Dokument und den Zeilen in einer relationalen Tabelle zu erstellen. Diese XPath-Ausdrücke identifizieren die Werte im XML-Dokument und ermitteln, wie diese Werte zu den Spalten der einzelnen Zeilen gehören. Die Daten des XML-Dokuments werden vom XML-Wrapper lediglich gelesen. Der XML-Wrapper führt keine Aktualisierung für diese Daten durch.

Beim Erstellen eines Kurznamens werden Optionen ausgewählt, die die Beziehung zwischen dem Kurznamen und dem XML-Dokument angeben. Kurznamen werden den XML-Dokumenten entweder fest zugeordnet oder mit Quellennamen, die Sie angeben.

Bei einer festen Zuordnung stellt der Kurzname Daten aus spezifischen XML-Dokumenten dar. Zu diesen XML-Dokumenten gehören:

### Eine lokale Datei

Sie geben eine XML-Datei als XML-Dokument an.

### Mehrere lokale Dateien in einem Verzeichnispfad

Sie geben einen Verzeichnispfad an, in dem sich mehrere XML-Dateien befinden. Die XML-Dateien in diesem Verzeichnispfad stellen dem Kurznamen die XML-Dokumentdaten zur Verfügung. Alle XML-Dateien müssen die gleiche Konfiguration aufweisen. Hat eine der XML-Dateien in dem Verzeichnis eine andere Konfiguration als der Kurzname, gibt der XML-Wrapper beim Verarbeiten der betreffenden XML-Datei Nullwerte zurück. Das Verzeichnis muss entweder lokal auf dem Server mit zusammengeschlossenen Datenbanken vorhanden sein oder es muss auf dieses Verzeichnis Zugriff über ein gemeinsam benutztes Dateisystem bestehen.

**Hinweis:** Beim Scannen des Verzeichnisses behält der Wrapper nur Dateien mit der Erweiterung `.xml` bei und führt auch nur für diese Dateien eine Syntaxanalyse durch. Der XML-Wrapper ignoriert alle anderen Dateien, einschließlich Dateien mit der Erweiterung `.txt` und `.xsd` sowie Dateien ohne Erweiterung.

Verwenden Sie die Option `FILE_PATH` der Anweisung `CREATE NICKNAME`, um Daten mit fester Größe aus einer Datei anzugeben. Verwenden Sie die Option `DIRECTORY_PATH`, um Daten mit fester Größe aus einem Verzeichnis anzugeben.

Werden die Quelldaten während der Ausführung der Abfrage angegeben, kann der Kurzname zur Darstellung von Daten aus beliebigen XML-Dokumentquellen verwendet werden, deren Schema der Kurznamendefinition entspricht. Zu diesen XML-Dokumenten gehören:

#### **Uniform Reference Identifiers (URIs)**

Eine ferne XML-Datei, auf die eine URI verweist, stellt dem Kurznamen die XML-Dokumentdaten zur Verfügung. (Geben Sie diese Dokumentquelle mit Hilfe der DOCUMENT-Kurznamenspaltenoption 'URI' an.)

#### **Relationale Spalten**

Spalten aus relationalen Tabellen, Sichten oder Kurznamen werden als Eingabe für das XML-Dokument verwendet. (Geben Sie diese Dokumentquelle mit Hilfe der DOCUMENT-Kurznamenspaltenoption 'COLUMN' an.)

**Datei** Eine einzelne Datei mit XML-Daten wird während der Ausführung der Abfrage als Eingabe zur Verfügung gestellt. (Geben Sie diese Dokumentquelle mit Hilfe der DOCUMENT-Kurznamenspaltenoption 'FILE' an.)

#### **Verzeichnis**

Mehrere XML-Dateien in einem angegebenen Verzeichnispfad stellen während der Ausführung der Abfrage die Daten zur Verfügung. (Geben Sie diese Dokumentquelle mit Hilfe der DOCUMENT-Kurznamenspaltenoption 'DIRECTORY' an.)

Mit der DOCUMENT-Spaltenoption wird angegeben, dass die Quelldaten bei Ausführung der Abfrage zur Verfügung gestellt werden. Geben Sie mit der Spalte DOCUMENT entweder URI, COLUMN, FILE oder DIRECTORY als Typ der XML-Dokumentquelle an.

Mit einer DOCUMENT-Spaltenoption kann nicht die Option FILE\_PATH oder DIRECTORY\_PATH angegeben werden.

Unabhängig vom verwendeten Datentyp (Daten mit festem Format oder Daten aus Quellennamen, die bei Ausführung der Abfrage angegeben werden) kann die Option STREAMING angegeben werden, die den XML-Wrapper dazu veranlasst, die XML-Dokumentdaten in Fragmente aufzuteilen. Der XML-Wrapper verarbeitet den daraus resultierenden XML-Datenstrom und extrahiert die angeforderten Informationen nach Abfragefragmenten. Der XML-Wrapper führt die Syntaxanalyse immer für ein Fragment zur Zeit aus. Dies reduziert die Gesamtspeicherbelegung. Je nach Speicherkapazität auf Ihrem Server kann sich jedoch die Verarbeitungszeit der gesamten Abfrage erhöhen. Daher sollte die Option STREAMING nur zur syntaktischen Analyse großer XML-Dokumente (Dokumente mit mindestens 50 MB) verwendet werden.

Sie haben auch die Möglichkeit, Werte für Kurznamenoptionen auszuwählen, die Ihnen bei der Optimierung von Abfragen helfen, bei denen große XML-Datenmengen oder Daten mit mehreren verschachtelten Elementen abgerufen werden. Zu diesen Optionen gehören:

- INSTANCE\_PARSE\_TIME
- XPATH\_EVAL\_TIME
- NEXT\_TIME

Sie können für diese Optionen Werte setzen, um die XML-Abfrage zu testen und zu optimieren. Diese Optionswerte steuern die erforderliche Verarbeitungszeit zum Suchen von Elementen und zum Analysieren der Syntax der Daten in den Zeilen des XML-Dokuments.

**Zugehörige Konzepte:**

- „Was ist XML?“ auf Seite 471
- „Die Aufwandsmodellfunktion für den XML-Wrapper“ auf Seite 481
- „Optimierungstipps für die XML-Aufwandsmodellfunktion“ auf Seite 488

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für XML-Datenquellen“ auf Seite 482

**Zugehörige Referenzen:**

- „Syntax der Anweisung CREATE NICKNAME - XML-Wrapper“ auf Seite 623
- „Anweisung CREATE NICKNAME - Beispiele für den XML-Wrapper“ auf Seite 482

**Die Aufwandsmodellfunktion für den XML-Wrapper**

Der XML-Wrapper stellt eine Aufwandsmodellfunktion zur Verfügung, um Abfragen für Kurznamen zu optimieren, die den XML-Quelldokumenten entsprechen.

Wenn Sie einen Kurznamen mit Hilfe der Anweisung CREATE NICKNAME erstellen, können Sie die folgenden Parameter als Kurznamenoptionswerte angeben, um die Aufwandsmodellfunktion zu unterstützen:

- INSTANCE\_PARSE\_TIME
- XPATH\_EVAL\_TIME

Sie können für diese Parameter die Standardwerte verwenden oder für diese Parameter Werte setzen, um Abfragen für die von Ihnen erstellten Stamm- und Nicht-Stammkurznamen zu optimieren.

Der Parameter INSTANCE\_PARSE\_TIME ist die erforderliche Zeit (in Millisekunden) zum Lesen und syntaktischen Analysieren eines zeilenproduzierenden Stammelements des Stammkurznamens (beispielsweise 'customers'), einschließlich aller enthaltenen zeilenproduzierenden Nicht-Stammelemente (beispielsweise alle Elemente, die den Bestellungen (orders), Zahlungen (payments) und Artikeln (items) der einzelnen Kunden (customer) entsprechen). Der XML-Wrapper erstellt im Speicher eine Struktur zur Darstellung dieser zeilenproduzierenden Stamm- und Nicht-Stammelemente.

Der Parameter XPATH\_EVAL\_TIME ist die erforderliche Zeit (in Millisekunden) zum Auswerten der XPath-Ausdrücke, mit deren Hilfe die Daten gesucht werden, die einer Zeile des Kurznamens entsprechen. Zu den ausgewerteten XPath-Ausdrücken gehören diejenigen, mit deren Hilfe die tatsächlichen Zielen gesucht werden, sowie diejenigen, mit deren Hilfe Spaltenwerte in diesen Zeilen gesucht werden.

**Zugehörige Konzepte:**

- „Was ist XML?“ auf Seite 471
- „Datenzuordnungen zwischen Kurznamen und XML-Dokumenten“ auf Seite 479
- „Optimierungstipps für die XML-Aufwandsmodellfunktion“ auf Seite 488

**Zugehörige Referenzen:**

- „Syntax der Anweisung CREATE NICKNAME - XML-Wrapper“ auf Seite 623
- „Anweisung CREATE NICKNAME - Beispiele für den XML-Wrapper“ auf Seite 482

## Registrieren von Kurznamen für XML-Datenquellen

Das Registrieren der Kurznamen für XML-Datenquellen ist Teil der umfangreichen Task des Hinzufügens von XML zu einem System zusammenschlossener Datenbanken. Sie müssen Kurznamen erstellen, die der Baumstruktur Ihrer XML-Datenquelle entsprechen. Übergeordnete Kurznamen (Elter) entsprechen dem Stamm der Baumstruktur. Untergeordnete Kurznamen (Kind) entsprechen den Elementen, die innerhalb des Elements für den übergeordneten Kurznamen verschachtelt sind.

### Voraussetzungen:

Die Codepage der Datenbank muss mit dem Zeichensatz der XML-Quellendateien übereinstimmen.

### Einschränkungen:

Namensbereiche werden nicht unterstützt.

### Vorgehensweise:

Um Kurznamen für XML-Datenquellen zu registrieren, geben Sie eine Anweisung `CREATE NICKNAME` aus.

Die nächste Task in dieser Tasksequenz ist das Erstellen von Sichten für zusammenschlossene Datenbanken für Nicht-Stammkurznamen (XML-Wrapper).

### Zugehörige Konzepte:

- „Datenzuordnungen zwischen Kurznamen und XML-Dokumenten“ auf Seite 479
- „Die Aufwandsmodellfunktion für den XML-Wrapper“ auf Seite 481

### Zugehörige Tasks:

- „Hinzufügen von XML zu einem System zusammenschlossener Datenbanken“ auf Seite 474
- „Erstellen von Sichten für zusammenschlossene Datenbanken für Nicht-Stammkurznamen (XML-Wrapper)“ auf Seite 487
- „Angabe von Kurznamenspalten für eine nicht relationale Datenquelle“ auf Seite 75

### Zugehörige Referenzen:

- „Syntax der Anweisung `CREATE NICKNAME` - XML-Wrapper“ auf Seite 623
- „Anweisung `CREATE NICKNAME` - Beispiele für den XML-Wrapper“ auf Seite 482

## Anweisung `CREATE NICKNAME` - Beispiele für den XML-Wrapper

Dieser Abschnitt enthält mehrere Beispiele, die zeigen, wie die Anweisung `CREATE NICKNAME` verwendet wird, um Kurznamen für den XML-Wrapper zu registrieren. Dieser Abschnitt enthält ein vollständiges Beispiel, das zeigt, wie über- und untergeordnete Kurznamen (Eltern und Kinder) erstellt werden, sowie Beispiele für spezifische Spaltenoptionen und Beispiele zur Verwendung von Sichten.

**Empfehlung:** Beim Angeben von XPATH-Spalten und Kurznamenoptionen in Ihren Abfragen sollten Sie nicht den Self-Operator bzw. den untergeordneten Operator



// verwenden. Der Self-Operator bzw. der untergeordnete Operator ist ein XPath-Operator, dessen Verwendung die Leistung des Servers mit zusammengesetzten Datenbanken beeinträchtigen kann.

### Vollständiges Beispiel:

Das folgende Beispiel zeigt, wie Kurznamen für XML-Datenquellen mit Hilfe der in Abb. 44 gezeigten XML-Beispieldatei erstellt werden.

---

```
<?xml version="1.0" encoding="UTF-8"?>
    <doc>
      <customer id='123'>
        <name>...</name>
        <address>...</address>
        ...
      <order>
        <amount>...</amount>
        <date>...</date>
        <item quant='12'>
          <name>...</name>
        </item>
        <item quant='4'>...</item>
        ...
      </order>
    <order>...</order>
    ...
    <payment>
      <number>...</number>
      <date>...</date>
    </payment>
    <payment>...</payment>
    ...
  </customer>
  <customer id='124'>...</customer>
</doc>
```

---

Abbildung 44. XML-Beispieldatei

### Der übergeordnete Kurzname:

Zunächst muss der übergeordnete Kurzname `customers` erstellt werden. Um diesen Kurznamen zu erstellen, setzen Sie die folgende Anweisung ab:

```
CREATE NICKNAME customers
(
  id          VARCHAR(5)  OPTIONS(XPATH './@id')
  name       VARCHAR(16)  OPTIONS(XPATH './name'),
  address    VARCHAR(30)  OPTIONS(XPATH './address/@street'),
  cid       VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(DIRECTORY_PATH '/home/db2user',
       XPATH '/doc/customer', STREAMING 'YES');
```

Mit dieser Anweisung wird der Kurzname `customers` über mehrere XML-Dateien im angegebenen Verzeichnispfad (`/home/db2user`) erstellt. Die Kurznamenoption `STREAMING` gibt an, dass die XML-Quellendaten getrennt vorliegen und nach Knoten verarbeitet werden (in diesem Beispiel nach Kundenstammdaten). Wird die Kurznamenoption `STREAMING` verwendet, speichert der Wrapper nicht das gesamte XML-Dokument. Stattdessen teilt der XML-Wrapper das Dokument in mehrere Abschnitte ein, die einzeln und nacheinander syntaktisch analysiert wer-



den. Die Kurznamenoption STREAMING sollte nur für große XML-Dokumente verwendet werden. Die Verwendung dieser Option beeinflusst die Leistung Ihrer Abfragen.

### Die untergeordneten Kurznamen:

Als Nächstes werden die untergeordneten Kurznamen für die Elemente orders, payments und items erstellt.

Um den untergeordneten Kurznamen orders zu erstellen, setzen Sie die folgende Anweisung ab:

```
CREATE NICKNAME orders
(
  amount  INTEGER      OPTIONS(XPATH './amount'),
  date    VARCHAR(10)  OPTIONS(XPATH './date'),
  oid     VARCHAR(16)  OPTIONS(PRIMARY_KEY 'YES'),
  cid     VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'CUSTOMERS'))
FOR SERVER xml_server
OPTIONS( XPATH './order');
```

Um den untergeordneten Kurznamen payments zu erstellen, setzen Sie die folgende Anweisung ab:

```
CREATE NICKNAME payments
(
  number  INTEGER      OPTIONS(XPATH './number'),
  date    VARCHAR(10)  OPTIONS(XPATH './date'),
  cid     VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'CUSTOMERS'))
FOR SERVER xml_server
OPTIONS( XPATH './payment');
```

Um den untergeordneten Kurznamen items zu erstellen, setzen Sie die folgende Anweisung ab:

```
CREATE NICKNAME items
(
  name     VARCHAR(20)  OPTIONS(XPATH './name'),
  quantity INTEGER      OPTIONS(XPATH './@quant'),
  oid     VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(FOREIGN_KEY 'ORDERS'))
FOR SERVER xml_server
OPTIONS( XPATH './item');
```

### Beispiele mit Spaltenoptionen:

Die nachstehenden Beispiel zeigen, wie die DOCUMENT-Spaltenoptionen beim Erstellen von Kurznamen eingeschlossen werden. Die Beispiele zeigen außerdem, wie diese Optionen in Abfragen verwendet werden.

Das folgende Beispiel der Anweisung CREATE NICKNAME zeigt die Verwendung der DOCUMENT-Spaltenoption 'FILE':

```
CREATE NICKNAME customers
(
  doc     VARCHAR(100)  OPTIONS(DOCUMENT 'FILE'),
  name    VARCHAR(16)   OPTIONS(XPATH './name'),
  address VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid     VARCHAR(16)  FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');
```

Anschließend können Sie die folgende Abfrage für den Kurznamen customers ausführen, indem Sie in der WHERE-Klausel die Speicherposition des XML-Dokuments angeben:

```
SELECT * FROM customers WHERE doc='/home/db2user/Customers.xml';
```

Das folgende Beispiel der Anweisung CREATE NICKNAME zeigt die Verwendung der DOCUMENT-Spaltenoption 'DIRECTORY':

```
CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'DIRECTORY'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid      VARCHAR(16)   FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');
```

Anschließend können Sie die folgende Abfrage für den Kurznamen customers ausführen:

```
SELECT name FROM customers WHERE doc = '/home/data/xml';
```

Mit dieser Abfrage werden die XML-Dokumente abgerufen, die sich im Verzeichnispfad /home/data/xml befinden, der in der WHERE-Klausel angegeben wurde.

Das folgende Beispiel der Anweisung CREATE NICKNAME zeigt die Verwendung der DOCUMENT-Kurznamenspaltenoption 'URI':

```
CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'URI'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid      VARCHAR(16)   FOR BIT DATA NOT NULL OPTIONS(PRIMARY_KEY 'YES'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/customer');
```

Anschließend können Sie die folgende Abfrage für den Kurznamen customers ausführen, um die XML-Daten von der fernen Speicherposition abzurufen:

```
SELECT * FROM customers WHERE doc = 'http://www.lg-mv.org/foo.xml';
```

Das folgende Beispiel der Anweisung CREATE NICKNAME zeigt die Verwendung der DOCUMENT-Kurznamenspaltenoption 'COLUMN':

```
CREATE NICKNAME emp
(
  doc      VARCHAR(500)  OPTIONS(DOCUMENT 'COLUMN')
  fname    VARCHAR(16)   OPTIONS(XPATH '@first'),
  lname    VARCHAR(16)   OPTIONS(XPATH '@last'))
FOR SERVER xml_server
OPTIONS(XPATH '/doc/name');
```

Anschließend können Sie die folgenden Abfragen für den Kurznamen emp ausführen, um die XML-Daten abzurufen:

```
SELECT * FROM emp WHERE doc = '<?xml version="1.0" encoding="UTF-8"?>
<doc>
  <title> employees </title>
  <name first="David" last="Marston"/>
  <name first="Donald" last="Leslie"/>
  <name first="Emily" last="Farmer"/>
  <name first="Myriam" last="Midy"/>
  <name first="Lee" last="Tran"/>
  <name first="Lili" last="Farmer"/>
  <name first="Sanjay" last="Kumar"/>
</doc>';
```

oder

```
SELECT * FROM emp WHERE doc = (SELECT * FROM xml_tab);
```

Die Tabelle `xml_tab` enthält eine Spalte, die mit den XML-Daten aufgefüllt ist.

### Beispiele für Sichten:

Die nachstehenden Beispiele zeigen, wie Sichten für Nicht-Rootkurznamen zum Beschreiben von XML-Quelldokumenten erstellt werden. In diesen Beispielen wird davon ausgegangen, dass die Kurznamen der in Abb. 45 dargestellten Beispieldatei zuvor als `customers`, `orders`, `payments` und `items` erstellt wurden.

---

```
<?xml version="1.0" encoding="UTF-8"?>
  <doc>
    <customer id='123'>
      <name>...</name>
      <address>...</address>
      ...
      <order>
        <amount>...</amount>
        <date>...</date>
        <item quant='12'>
          <name>...</name>
        </item>
        <item quant='4'>...</item>
        ...
      </order>
      <order>...</order>
      ...
      <payment>
        <number>...</number>
        <date>...</date>
      </payment>
      <payment>...</payment>
      ...
    </customer>
    <customer id='124'>...</customer>
  </doc>
```

---

Abbildung 45. XML-Beispieldatei

Das folgende Beispiel zeigt, wie eine Sicht für den Nicht-Rootkurznamen `order` erstellt wird:

```
CREATE VIEW order_view AS
SELECT o.amount, o.date, o.oid, c.cid
FROM customers c, orders o
WHERE c.cid = o.cid;
```

Das folgende Beispiel zeigt, wie eine Sicht für den Nicht-Rootkurznamen `payment` erstellt wird:

```
CREATE VIEW payment_view AS
  SELECT p.number, p.date, c.cid
FROM customers c, payments p
WHERE c.cid = p.cid;
```

Das folgende Beispiel zeigt, wie eine Sicht für den Nicht-Rootkurznamen `item` erstellt wird:

```
CREATE VIEW item_view AS
  SELECT i.quantity, i.name, o.oid
  FROM customers c, orders o, items i
  WHERE c.cid = o.cid AND o.oid = i.oid;
```

Abfragen, die an diese Sichten übergeben werden, werden einwandfrei verarbeitet, da der Verknüpfungspfad zum Stammverzeichnis (Root) vorhanden ist.

In der folgenden Abfrage beispielsweise werden die Beträge (amounts) der Bestellungen (orders) und Zahlungen (payments) für Kunden zusammen nach demselben Datum ausgewählt:

```
SELECT o.amount, p.amount
  FROM order_view o, payment_view p
  WHERE p.date = o.date AND
        p.cid = o.cid;
```

#### Zugehörige Tasks:

- „Registrieren von Kurznamen für XML-Datenquellen“ auf Seite 482

#### Zugehörige Referenzen:

- Anhang G, „Kurznamenoptionen für Systeme zusammengesetzter Datenbanken“, auf Seite 671
- Anhang F, „Kurznamenoptionen für Systeme zusammengesetzter Datenbanken“, auf Seite 661
- „Syntax der Anweisung CREATE NICKNAME - XML-Wrapper“ auf Seite 623

## Erstellen von Sichten für zusammengesetzte Datenbanken für Nicht-Stammkurznamen (XML-Wrapper)

Das Erstellen von Sichten für zusammengesetzte Datenbanken für Nicht-Stammkurznamen (XML-Wrapper) ist Teil der umfangreicheren Task des Hinzufügens von XML zu einem System zusammengesetzter Datenbanken.

Sie können Sichten für zusammengesetzte Datenbanken über die Hierarchie der Kurznamen definieren, die ein XML-Dokument beschreiben. Das Definieren von Sichten für zusammengesetzte Datenbanken gewährleistet die ordnungsgemäße Ausführung von Abfragen, die Elemente einer XML-Kurznamenhierarchie miteinander verknüpfen (ausschließlich des Stammkurznamen und Abfragen, die andere Spalten als die spezifischen Spalten PRIMARY\_KEY und FOREIGN\_KEY verknüpfen).

#### Vorgehensweise:

Führen Sie folgende Schritte aus, um Sichten für zusammengesetzte Datenbanken zu definieren, die alle erforderlichen Vergleichselemente und einen vollständigen Pfad zum Stammverzeichnis enthalten:

1. Definieren Sie eine Sicht für jeden Nicht-Stammkurznamen als Verknüpfung aller Kurznamen in dem Pfad zum Stamm.
2. Setzen Sie die Verknüpfungsvergleichselemente in der WHERE-Klausel über die Spalten PRIMARY\_KEY und FOREIGN\_KEY.
3. Integrieren Sie alle Spalten des Nicht-Stammkurznamen in die SELECT-Liste, mit Ausnahme der Spalte, die mit der Kurznamenoption FOREIGN\_KEY ausgewiesen ist.

4. Integrieren Sie die Spalte des übergeordneten Kurznamens, die mit der Option PRIMARY\_KEY ausgewiesen ist, in die SELECT-Liste.

**Zugehörige Referenzen:**

- „Anweisung CREATE NICKNAME - Beispiele für den XML-Wrapper“ auf Seite 482

## Optimierungstipps für die XML-Aufwandsmodellfunktion

Die Aufwandsmodellfunktion des XML-Wrappers hilft bei der Optimierung von Abfragen für die von Ihnen erstellten Kurznamen.

Die Aufwandsmodellfunktion verwendet die folgenden Parameter der Anweisung CREATE NICKNAME:

- INSTANCE\_PARSE\_TIME
- XPATH\_EVAL\_TIME

Wenn Sie die Anweisung CREATE NICKNAME absetzen, um einen Kurznamen für eine XML-Datenquelle zu registrieren, können Sie Werte für diese Parameter angeben.

Die Aufwandsmodellfunktion verwendet diese Parameterwerte bei der Ermittlung der erforderlichen Zeit für die Syntaxanalyse der Daten in den einzelnen Zeilen eines XML-Quelldokuments sowie für die Auswertung des XPath-Ausdrucks des Kurznamens.

Sie können für diese Parameter die Standardwerte verwenden. Wenn Sie jedoch Abfragen für umfangreiche oder komplexe XML-Quellenstrukturen für die von Ihnen erstellten Kurznamen optimieren wollen, sollten Sie folgendes Beispiel als Richtlinie verwenden:

**Beispiel zum Optimieren einer umfangreichen Abfrage:**

Nehmen Sie an, Ihr XML-Dokument hat ein relationales Schema mit den folgenden vier Kurznamen:

- customers (Kunden)
- orders (Bestellungen)
- payments (Zahlungen)
- items (Artikel)

Nehmen Sie weiterhin an, dass der Kurzname 'customers' der Stammkurzname (Root) ist.

Führen Sie für jeden Kurznamen Abfragen aus. Führen Sie jede Abfrage für ein in Ihrer Umgebung typisches Beispiel der XML-Daten aus.

Beispiel:

```
SELECT * from customers;  
SELECT * from orders;  
SELECT * from payments;  
SELECT * from items;
```

Notieren Sie die erforderliche Zeit (in Millisekunden) zum Ausführen der einzelnen Abfragen, indem Sie den Befehl **db2batch** oder einen äquivalenten Befehl bzw.

ein äquivalentes Dienstprogramm verwenden. (Mit Hilfe des Befehls **db2batch** können Sie eine Ausgabedatei erstellen, die die erforderlichen Zeiten für die Ausführung der Abfragen enthält.) Notieren Sie außerdem die Anzahl der zurückgegebenen Tuples.

Verwenden Sie für jeden Kurznamen die folgenden Formeln zum Ermitteln der optimalen Werte für die Parameter `INSTANCE_PARSE_TIME` und `XPATH_EVAL_TIME`:

$$\text{INSTANCE\_PARSE\_TIME} = (75\% \times \text{Laufzeit der Abfrage SELECT } *) \div \text{Anzahl der zurückgegebenen Tuples}$$
$$\text{XPATH\_EVAL\_TIME} = (25\% \times \text{Laufzeit der Abfrage SELECT } *) \div \text{Anzahl der zurückgegebenen Tuples}$$

Verwenden Sie für den Stammkurznamen (in diesem Beispiel `customers`) die errechneten Werte für die Parameter `INSTANCE_PARSE_TIME` und `XPATH_EVAL_TIME`.

Verwenden Sie für Nicht-Stammkurznamen (in diesem Beispiel `orders`, `payments` und `items`) nur den errechneten Wert für den Parameter `XPATH_EVAL_TIME`. Der Wert des Parameters `INSTANCE_PARSE_TIME` gilt nicht für Nicht-Stammkurznamen.

Sie können diese Formeln beim Optimieren Ihrer Abfragen als Richtlinie verwenden. Die optimalen Werte für diese Parameter hängen auch von der Komplexität Ihrer XML-Quelldokumente und von der Geschwindigkeit des verwendeten Prozessors ab.

#### Zugehörige Konzepte:

- „Was ist XML?“ auf Seite 471
- „Datenzuordnungen zwischen Kurznamen und XML-Dokumenten“ auf Seite 479
- „Die Aufwandsmodellfunktion für den XML-Wrapper“ auf Seite 481

#### Zugehörige Referenzen:

- „db2batch - Benchmark Tool Command“ in *Command Reference*

---

## XML-Datenquelle - Beispielabfragen

Dieser Abschnitt enthält mehrere Beispielabfragen, in denen die Kurznamen `'customers'`, `'orders'` und `'items'` verwendet werden. Diese Kurznamen wurden zuvor jeweils mit Hilfe der entsprechenden Anweisung `CREATE NICKNAME` registriert. Mit der folgenden Abfrage werden alle Namen von Kunden (`customers`) angezeigt:

```
SELECT name FROM customers;
```

Mit der folgenden Abfrage werden alle Einträge angezeigt, in denen der Name des Kunden `'Chang'` lautet:

```
SELECT * FROM customers
WHERE name='Chang';
```

Mit der folgenden Abfrage werden für jede Bestellung (`order`) jedes Kunden (`customer`) der Name (`name`) und die Beträge (`amounts`) angezeigt:

```
SELECT c.name, o.amount FROM customers c, orders o
WHERE c.cid=o.cid;
```

Sie müssen die Verknüpfung 'c.cid=o.cid' für die Elter-Kind-Beziehung zwischen dem Kurznamen 'customers' und dem Kurznamen 'orders' angeben.

Mit der folgenden Abfrage werden die Adressen der Kunden, die Beträge der Bestellungen und die Namen der Artikel (item) für alle Bestellungen und Artikel der jeweiligen Kunden ausgewählt:

```
SELECT c.address, o.amount, i.name FROM customers c, orders o, items i
WHERE c.cid=o.cid AND o.oid=i.oid;
```

Sie müssen die beiden Verknüpfungen angeben, um die Elter-Kind-Beziehungen aufrecht zu erhalten.

Die folgenden Beispiele zeigen, wie Abfragen geschrieben werden, bei denen ein Kurzname verwendet wird, der keine Kurznamenoption FILE\_PATH, aber eine Spaltenoption DOCUMENT angibt. Nachfolgend wird die entsprechende Anweisung CREATE NICKNAME gezeigt, die zum Erstellen des Kurznamens customers verwendet wird:

```
CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'FILE'),
  name     VARCHAR(16)   OPTIONS(XPATH './name'),
  address  VARCHAR(30)   OPTIONS(XPATH './address/@street'),
  cid     VARCHAR(16)   OPTIONS(PRIMARY_KEY 'YES')
  FOR SERVER xml_server
  OPTIONS(XPATH '/doc/customer');
```

Mit der folgenden Abfrage werden alle Daten aus der XML-Datei Customers.xml mit dem Dateipfad /home/db2user/Customers.xml ausgewählt:

```
SELECT * FROM customers
WHERE doc='/home/db2user/Customers.xml';
```

Mit der folgenden Abfrage werden die Namen von Kunden und die Datumsangaben (dates) ihrer Bestellungen über einen Betrag von mehr als 1000 aus der XML-Datei Customers.xml ausgewählt:

```
SELECT c.name, o.date FROM customers c, orders o
WHERE c.doc='/home/db2user/Customers.xml' AND o.amount > 1000;
```

Der Dateipfad /home/db2user/Customers.xml gibt die Speicherposition der Datei Customers.xml an.

#### Zugehörige Referenzen:

- „Syntax der Anweisung CREATE NICKNAME - XML-Wrapper“ auf Seite 623
- „Anweisung CREATE NICKNAME - Beispiele für den XML-Wrapper“ auf Seite 482

---

## Nachrichten für den XML-Wrapper

In diesem Abschnitt werden Nachrichten beschrieben, die beim Arbeiten mit dem Wrapper für XML auftreten können. Weitere Informationen zu Nachrichten finden Sie im Handbuch zu *DB2 Fehlernachrichten*.

*Tabelle 92. Vom XML-Wrapper ausgegebene Nachrichten*

Fehlercode	Nachricht	Erläuterung
SQL0405N	Das numerische Literal " <code>&lt;spaltenname&gt;</code> " ist nicht gültig, da sein Wert außerhalb des zulässigen Bereichs liegt.	Das angegebene numerische Literal befindet sich nicht im zulässigen Bereich. Überprüfen Sie den Datentyp der Spalte in der Anweisung <code>CREATE NICKNAME</code> .
SQL0408N	Ein Wert ist mit dem Datentyp seines Zuordnungsziels nicht kompatibel. Der Name des Ziels ist " <code>&lt;spaltenname&gt;</code> ".	Der Datentyp des Wertes, der der Spalte zugeordnet werden soll, ist mit dem deklarierten Datentyp des Zuordnungsziels nicht kompatibel. Überprüfen Sie den Datentyp der Spalte in der Anweisung <code>CREATE NICKNAME</code> .
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Fehler beim Erstellen eines Wrapperobjekts".)	Beim Erstellen eines neuen Oberflächenobjekts ist ein Fehler aufgetreten. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: " <code>&lt;xerces_xalan_fehlermeldung&gt;</code> ".)	Während des Aufrufs einer Xerces- oder Xalan-Funktion ist ein Fehler aufgetreten. Überprüfen Sie das XML-Dokument. Ist das Dokument richtig strukturiert, lesen Sie die Xalan-Dokumentation, um weitere Informationen zur Fehlermeldung zu erhalten.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "XalanDOMException: Der Ausnahmecode ist <code>&lt;ausnahmecode&gt;</code> ".)	Die Ausnahmecodebedingung 'XalanDOMException' trat auf. Weitere Informationen zu diesem Ausnahmecode finden Sie in der Xalan-Dokumentation.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "XMLException: <code>&lt;ausnahmecodefehlermeldung&gt;</code> ".)	Die Ausnahmecodebedingung XMLException trat auf. Weitere Informationen zu diesem Ausnahmecode finden Sie in der Xalan-Dokumentation.



Tabelle 92. Vom XML-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "XSLEException: <ausnahmebedingungsfehlernachricht>".)	Die Ausnahmebedingung XSLEException trat auf. Weitere Informationen zu diesem Ausnahmebedingungscode finden Sie in der Xalan-Dokumentation.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "SAXParseException: <ausnahmebedingungsfehlernachricht>".)	Die Ausnahmebedingung SAXParseException trat auf. Weitere Informationen zu diesem Ausnahmebedingungscode finden Sie in der Xalan-Dokumentation.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Fehler beim Abrufen des Knotenwerts".)	Xalan versuchte, auf einen ungültigen Knoten zuzugreifen. Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Fehler bei Syntaxanalyse des XML-Dokuments".)	Bei der Syntaxanalyse des XML-Dokuments trat ein Fehler auf. Überprüfen Sie das XML-Dokument.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Fehler beim Abrufen von Stammelement aus XML-Dokument".)	Nach der Syntaxanalyse des XML-Dokuments versuchte Xalan das Stammelement abzurufen. Der Abruf schlug jedoch fehl. Überprüfen Sie das XML-Dokument.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Unspezifische Ausnahmebedingung bei Auswertung des XPath-Ausdrucks".)	Bei der Auswertung eines XPath-Ausdrucks generierte Xalan eine unspezifische Ausnahmebedingung. Überprüfen Sie das XML-Dokument und lesen Sie die Xalan-Dokumentation.

Tabelle 92. Vom XML-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Unspezifische Ausnahmebedingung beim Abrufen des Knotenwerts".)	Beim Abrufen eines Knotenwerts generierte Xalan eine unspezifische Ausnahmebedingung. Überprüfen Sie das XML-Dokument und lesen Sie die Xalan-Dokumentation.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Unspezifische Ausnahmebedingung bei Syntaxanalyse des Eingabedokuments".)	Bei der Syntaxanalyse des XML-Dokuments generierte Xalan eine unspezifische Ausnahmebedingung. Überprüfen Sie das XML-Dokument und lesen Sie die Xalan-Dokumentation.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Fehler beim Auswerten der Kardinalität".)	Wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "<SOAP_fehlernachricht>".)	Die SOAP-Bibliothek gab einen Fehler aus. Falls Sie den Fehler in der SQL-Anweisung nicht auflösen können, wenden Sie sich an die IBM Softwareunterstützung.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Ungültige URI".)	Der Wrapper kann nicht auf die angegebene URL zugreifen. Überprüfen Sie, ob ein Zugriff auf die URL möglich ist.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Ungültiger XML-Dokumentinhalt".)	Der Inhalt des XML-Dokuments ist ungültig. Überprüfen Sie, ob das Dokument richtig strukturiert ist.
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Ungültige SOAP-Rahmenanweisung".)	Die SOAP-Rahmenanweisung ist ungültig. Überprüfen Sie die Syntax und den Inhalt.

Tabelle 92. Vom XML-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0901N	Die SQL-Anweisung schlug aufgrund eines nicht schwerwiegenden (nicht kritischen) Systemfehlers fehl. Nachfolgende SQL-Anweisungen können verarbeitet werden. (Ursache: "Fehler bei Speicherzuordnung.")	Bei der Speicherzuordnung trat ein Fehler auf.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Falsches DATE-Format."	Das Format eines Datumswertes in dem XML-Dokument ist nicht korrekt. Das gültige Format für Datumswerte ist jjjj-mm-tt. Überprüfen Sie das XML-Dokument.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Spaltendatentyp nicht unterstützt".	Eine Kurznamenspalte weist einen nicht unterstützten Datentyp auf. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "TYPE-Klausel nicht unterstützt".	Die Anweisung CREATE SERVER enthält eine TYPE-Klausel. Diese Klausel wird vom XML-Wrapper nicht unterstützt. Entfernen Sie die Klausel.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "VERSION-Klausel nicht unterstützt".	Die Anweisung CREATE SERVER enthält eine VERSION-Klausel. Diese Klausel wird vom XML-Wrapper nicht unterstützt. Entfernen Sie die Klausel.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Ungültige Verwendung eines Vergleichselements mit Spalte DOCUMENT".	Die Abfrage enthält ein Vergleichselement mit falschen Operanden. Überprüfen Sie die Vergleichselemente in der Abfrage.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Ungültige Verwendung eines Vergleichselements mit Spalte FOREIGN_KEY".	Die Abfrage enthält ein Vergleichselement mit falschen Operanden. Überprüfen Sie die Vergleichselemente in der Abfrage.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Ungültige Verwendung eines Vergleichselements mit Spalte PRIMARY_KEY".	Die Abfrage enthält ein Vergleichselement mit falschen Operanden. Überprüfen Sie die Vergleichselemente in der Abfrage.

Tabelle 92. Vom XML-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen XPATH und DOCUMENT sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen XPATH und FOREIGN_KEY sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen XPATH und PRIMARY_KEY sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen DOCUMENT und FOREIGN_KEY sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen DOCUMENT und PRIMARY_KEY sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen FOREIGN_KEY und PRIMARY_KEY sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Spaltenoption fehlt".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Spaltenoption DOCUMENT ist nicht eindeutig".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.

Tabelle 92. Vom XML-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Spaltenoption FOREIGN_KEY ist nicht eindeutig".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Spaltenoption PRIMARY_KEY ist nicht eindeutig".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie die Syntax der Anweisung.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Ungültiger Wert für Option DOCUMENT".	Der in der Anweisung CREATE NICKNAME angegebene Wert der Option DOCUMENT ist nicht gültig. Der Wert muss FILE lauten. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Ungültiger Wert für Option PRIMARY_KEY".	Der in der Anweisung CREATE NICKNAME angegebene Wert der Option PRIMARY_KEY ist nicht gültig. Der Wert muss YES lauten. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Ungültiger Wert für Option FOREIGN_KEY".	Der in der Anweisung CREATE NICKNAME angegebene Wert der Option FOREIGN_KEY ist nicht gültig. Der Wert stimmt mit keinem Kurznamen eines übergeordneten Elements überein. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen FILE_PATH und DOCUMENT sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Die Optionen FILE_PATH und DOCUMENT können nicht gleichzeitig angegeben werden. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen FILE_PATH und SOAP sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Die Optionen FILE_PATH und SOAP können nicht gleichzeitig angegeben werden. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen DIRECTORY_PATH und SOAP sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Die Optionen DIRECTORY_PATH und SOAP können nicht gleichzeitig angegeben werden. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.

Tabelle 92. Vom XML-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen FILE_PATH und DIRECTORY_PATH sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Die Optionen FILE_PATH und DIRECTORY_PATH können nicht gleichzeitig angegeben werden. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen VALIDATE und STREAMING sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Die Optionen VALIDATE und STREAMING können nicht gleichzeitig angegeben werden. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen FILE_PATH und FOREIGN_KEY sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Die Optionen FILE_PATH und FOREIGN_KEY können nicht gleichzeitig angegeben werden. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Die Optionen DIRECTORY_PATH und FOREIGN_KEY sind nicht kompatibel".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Die Optionen DIRECTORY_PATH und FOREIGN_KEY können nicht gleichzeitig angegeben werden. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Ungültiger Wert für Option XPATH bei aktivierter Funktion STREAMING".	Der XPATH-Ausdruck des Kurznamens ist ungültig, wenn die Funktion STREAMING aktiviert ist. Überprüfen Sie die Option XPATH auf ungültige Werte, wie z. B. /, ./ und //.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "XML-Datei kann nicht gelesen werden".	Der in der Anweisung CREATE NICKNAME oder in der Abfrage angegebene Dateipfad ist ungültig. Die angegebene Datei existiert nicht. Überprüfen Sie die Anweisung CREATE NICKNAME und die Abfrage.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Verzeichnis kann nicht geöffnet werden".	Der in der Anweisung CREATE NICKNAME oder in der Abfrage angegebene Verzeichnispfad ist ungültig. Das angegebene Verzeichnis existiert nicht. Überprüfen Sie die Anweisung CREATE NICKNAME und die Abfrage.



Tabelle 92. Vom XML-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Verweis auf XML-Daten fehlt".	Die Anweisung CREATE NICKNAME muss einen Verweis auf die XML-Daten enthalten. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Spaltenoption DOCUMENT mit Wert 'SOAP' fehlt".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Überprüfen Sie den Wert der Option DOCUMENT. Der Wert muss SOAP lauten.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Option SOAP fehlt".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Die Option SOAP muss angegeben werden.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "INSTANCE_PARSE_TIME nur für Stammkurznamen".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Sie können nur für Stammkurznamen einen Wert für INSTANCE_PARSE_TIME angeben. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Option VALIDATE nur für Stammkurznamen".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Sie können die Option VALIDATE nur auf YES setzen, wenn der angegebene Kurzname ein Stammkurzname ist. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1822N	Es wurde ein unerwarteter Fehlercode "<tracepunkt>" von der Datenquelle "XML-Wrapper" empfangen. Zugeordneter Text und Token sind "Option STREAMING nur für Stammkurznamen".	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Sie können die Option STREAMING nur auf YES setzen, wenn der angegebene Kurzname ein Stammkurzname ist. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1823N	Für den Datentyp "<datentypname>" gibt es vom Server "<servername>" keine Datentypzuordnung.	Die Anweisung CREATE NICKNAME wurde nicht richtig angegeben. Ein Spaltendatentyp ist ungültig. Überprüfen Sie die Syntax der Anweisung CREATE NICKNAME.
SQL1881N	"<optionsname>" ist keine gültige Option des Typs "<optionstyp>" für "<objektname>".	Die angegebene Option existiert möglicherweise nicht oder ist für diese Datenquelle möglicherweise nicht gültig. Überprüfen Sie die Anweisung CREATE NICKNAME.

Tabelle 92. Vom XML-Wrapper ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL1881N	"DIRECTORY_PATH" ist keine gültige Option "NICKNAME" für "<objektname>".	Der in der Anweisung CREATE NICKNAME angegebene Wert der Option DIRECTORY_PATH ist nicht gültig. Das angegebene Verzeichnis muss ein Stammverzeichnis sein. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1882N	Die Option "VALIDATE" der Art "kurzname" kann nicht auf "<optionswert>" für "<objektname>" eingestellt werden.	Der in der Anweisung CREATE NICKNAME angegebene Wert der Option VALIDATE ist nicht gültig. Dieser Wert muss entweder YES oder NO lauten. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1882N	Die Option "STREAMING" der Art "kurzname" kann nicht auf "<optionswert>" für "<objektname>" eingestellt werden.	Der in der Anweisung CREATE NICKNAME angegebene Wert der Option STREAMING ist nicht gültig. Dieser Wert muss entweder YES oder NO lauten. Überprüfen Sie die Anweisung CREATE NICKNAME.
SQL1883N	"<optionsname>" ist eine erforderliche Option der Art "<optionstyp>" für "<objektname>".	Eine erforderliche DB2-Option wurde nicht angegeben. Überprüfen Sie die Anweisung CREATE NICKNAME.

**Zugehörige Konzepte:**

- „Nachrichten - Einführung“ in *Fehlernachrichten Band 1*

**Zugehörige Referenzen:**

- „SQLSTATE-Nachrichten“ in *Fehlernachrichten Band 2*





---

## Teil 4. Benutzerdefinierte Funktionen



---

## Kapitel 25. Benutzerdefinierte Life Sciences-Funktionen

In diesem Kapitel wird erläutert, was die benutzerdefinierten Life Sciences-Funktionen sind, wie sie einem System zusammengeschlossener Datenbanken hinzugefügt und in Abfragen verwendet werden.

---

### Benutzerdefinierte Life Sciences-Funktionen - Übersicht

#### Benutzerdefinierte Life Sciences-Funktionen - Übersicht

Die benutzerdefinierten Life Sciences-Funktionen stellen Ihnen häufig verwendete Algorithmen zur Datenanalyse zur Verfügung.

Die benutzerdefinierten Life Sciences-Funktionen verwenden zur Darstellung von Aminosäuren und Nukleotiden die standardmäßigen Einzelbuchstabencodes sowie die IUPAC-IUB-Ambiguitätscodes.

Die benutzerdefinierten Life Sciences-Funktionen werden mit der Komponente für benutzerdefinierte Life Sciences-Funktionen der nicht relationalen Wrapper installiert. Nach der Installation der benutzerdefinierten Life Sciences-Funktionen müssen Sie diese Funktionen registrieren.

Um Konflikte mit Namensbereichen zu vermeiden, werden alle benutzerdefinierten Life Sciences-Funktionen im Schema DB2LS registriert.

#### Zugehörige Konzepte:

- „DB2 Information Integrator - Nicht relationale Wrapper“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „DB2 Information Integrator - Relationale Wrapper“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*

#### Zugehörige Tasks:

- „Registrieren von benutzerdefinierten Life Sciences-Funktionen“ auf Seite 505
- „Inaktivieren der benutzerdefinierten Life Sciences-Funktionen“ auf Seite 507

#### Zugehörige Referenzen:

- „Bibliotheksdateien der benutzerdefinierten Life Sciences-Funktionen“ auf Seite 503
- „Benutzerdefinierte Life Sciences-Funktionen nach Funktionskategorie“ auf Seite 504

#### Bibliotheksdateien der benutzerdefinierten Life Sciences-Funktionen

Für einige der benutzerdefinierten Funktionen in DB2 Information Integrator sind Bibliotheksdateien erforderlich. Diese Bibliotheksdateien sind erforderlich, wenn Sie die benutzerdefinierten Funktionen auf dem Server mit zusammengeschlossenen Datenbanken registrieren.

Wenn Sie DB2 Information Integrator für nicht relationale Datenquellen installieren, werden die folgenden Bibliotheksdateien für benutzerdefinierte Funktionen auf dem Server mit zusammengeschlossenen Datenbanken installiert:

**Bibliotheken der benutzerdefinierten Life Sciences-Funktionen:**

*Tabelle 93. Speicherpositionen und Dateinamen der Bibliotheken der benutzerdefinierten Life Sciences-Funktionen*

Funktionstyp	Betriebssystem	Verzeichnispfad	Bibliotheksdateiname
Benutzerdefinierte Life Sciences-Funktionen	AIX	/SQLLIB/function	libdb2lsudfs.a
Benutzerdefinierte Life Sciences-Funktionen	HP-UX	/SQLLIB/function	libdb2lsudfs.sl
Benutzerdefinierte Life Sciences-Funktionen	Linux	/SQLLIB/function	libdb2lsudfs.so
Benutzerdefinierte Life Sciences-Funktionen	Solaris	/SQLLIB/function	libdb2lsudfs.so
Benutzerdefinierte Life Sciences-Funktionen	Windows	%DB2PATH%\bin	db2lsudfs.dll

Der Standardverzeichnispfad unter Windows lautet C:\Program Files\IBM\SQLLIB. %DB2PATH% ist die Umgebungsvariable, mit der der Verzeichnispfad angegeben wird, in dem DB2 Information Integrator unter Windows installiert ist.

**Bibliothek der benutzerdefinierten Funktion 'LSGeneWise':**

Für die benutzerdefinierte Funktion 'LSGeneWise' ist eine eigene Bibliothek erforderlich.

*Tabelle 94. Speicherposition und Dateiname der Bibliothek der Funktion 'LSGeneWise'*

Funktionstyp	Betriebssystem	Verzeichnispfad	Bibliotheksdateiname
Funktion 'LSGeneWise'	UNIX	/SQLLIB/lib	libdb2lsSTgenewise.a

**Zugehörige Konzepte:**

- „Benutzerdefinierte Life Sciences-Funktionen - Übersicht“ auf Seite 503

## Benutzerdefinierte Life Sciences-Funktionen nach Funktionskategorie

Tabelle 95 enthält eine Liste der benutzerdefinierten Life Sciences-Funktionen nach ihrer jeweiligen Funktionskategorie mit kurzer Beschreibung.

*Tabelle 95. Benutzerdefinierte Life Sciences-Funktionen*

Funktionskategorie	Benutzerdefinierte Funktionen	Beschreibung
Rückübersetzung	LSPep2AmbNuc, LSPep2ProbNuc	Wandelt eine Aminosäuresequenz in eine Nukleotidsequenz um.

Tabelle 95. Benutzerdefinierte Life Sciences-Funktionen (Forts.)

Funktionskategorie	Benutzerdefinierte Funktionen	Beschreibung
Syntaktische Analyse von Definitionszeilen	LSDefineParse	Führt eine syntaktische Analyse von Elementen einer Definitionszeile durch, wie sie beispielsweise vom BLAST-Wrapper zurückgegeben wird oder in Datendateien im FASTA-Format vorliegt.
Allgemeine Mustererkennung	LSPatternMatch, LSPrositePattern	Ermittelt relevante Bereiche in einer angegebenen Zeichenfolge, wie beispielsweise Nukleotid- oder Peptidsequenzen.
GeneWise	LSGeneWise	Führt ein Alignment einer Proteinsequenz mit einer Genomsequenz durch.
Motive	LSMultiMatch, LSMultiMatch3, LSBarCode	Sucht übereinstimmende Muster in Nukleotid- oder Aminosäuresequenzen.
Umkehrung	LSRevNuc, LSRevPep, LSRevComp	Kehrt Nukleotid- oder Aminosäuresequenzen um.
Übersetzung	LSNuc2Pep, LSTransAllFrames	Wandelt eine Nukleotidsequenz in eine Peptidsequenz um.

**Zugehörige Konzepte:**

- „Benutzerdefinierte Life Sciences-Funktionen - Übersicht“ auf Seite 503

**Zugehörige Tasks:**

- „Registrieren von benutzerdefinierten Life Sciences-Funktionen“ auf Seite 505
- „Inaktivieren der benutzerdefinierten Life Sciences-Funktionen“ auf Seite 507

## Registrieren von benutzerdefinierten Life Sciences-Funktionen

Bevor Sie die benutzerdefinierten Life Sciences-Funktionen verwenden können, müssen Sie diese Funktionen registrieren.

**Voraussetzungen:**

Die Komponente für benutzerdefinierte Life Sciences-Funktionen der nicht relationalen Wrapper muss zusammen mit DB2 Information Integrator installiert werden.

**Vorgehensweise:**

Sie können die benutzerdefinierten Life Sciences-Funktionen mit Hilfe des Befehls 'enable\_LSFuctions' registrieren.

- Auf Servern mit zusammengeschlossenen Datenbanken unter Windows NT befindet sich dieser Befehl im `sqllib\bin` Verzeichnis.
- Auf Servern mit zusammengeschlossenen Datenbanken unter AIX befindet sich dieser Befehl im Verzeichnis `sqllib/bin`.

Die Syntax für den Befehl 'enable\_LSFuctions' lautet wie folgt:

```
enable_LSFuctions -n dbname -u benutzer-id -p kennwort [-force]
```

**dbname**

Der Name der zusammengesetzten Datenbank, in der Sie die Funktionen registrieren.

**benutzer-id**

Eine gültige Benutzer-ID für die zusammengesetzte Datenbank.

**kennwort**

Ein gültiges Kennwort für die Benutzer-ID.

**force** Eine Markierung, mit der die Funktionen entfernt und erneut registriert werden können. Mit Hilfe dieser Markierung können Sie die Funktionen erneut registrieren, falls sie beschädigt oder versehentlich gelöscht werden.

Mit dem Befehl 'enable\_LSFfunctions' werden alle benutzerdefinierten Life Sciences-Funktionen in der zusammengesetzten Datenbank registriert. Die Registrierung der Funktionen erfolgt mit dem Schemanamen DB2LS.

Das folgende Beispiel zeigt die Ausgabe, die nach Absetzen des Befehls 'enable\_LSFfunctions' zurückgegeben wird:

```
C:> enable_LSFfunctions -n federateddb -u db2admin -p db2admin
```

```
(0) Life Sciences-Funktionen wurden gefunden
-- Erstellen von Life Sciences-Funktionen ...
Life Sciences-Funktionen erfolgreich erstellt.
```

\*\*\* Das Bereinigen des Systems kann einige Sekunden dauern. Bitte warten Sie .....

Das folgende Beispiel zeigt die Ausgabe, die nach Absetzen des Befehls 'enable\_LSFfunctions' zurückgegeben wird, wenn Sie die Markierung 'force' verwenden und die Funktionen bereits registriert sind:

```
C:> enable_LSFfunctions -n federateddb -u db2admin -p db2admin -force
```

```
(21) Life Sciences-Funktionen wurden gefunden

Life Sciences-Funktionen sind bereits vorhanden ...
Erneute Installation von Life Sciences-Funktionen ...
-- Löschen von Life Sciences-Funktionen ...
Life Sciences-Funktionen erfolgreich gelöscht.
-- Erstellen von Life Sciences-Funktionen ...
Life Sciences-Funktionen erfolgreich erstellt.
```

\*\*\* Das Bereinigen des Systems kann einige Sekunden dauern. Bitte warten Sie .....

**Zugehörige Konzepte:**

- „Benutzerdefinierte Life Sciences-Funktionen - Übersicht“ auf Seite 503

**Zugehörige Tasks:**

- „Hinzufügen von relationalen Wrappern, nicht relationalen Wrappern und benutzerdefinierten Funktionen zu DB2 Information Integrator-System“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Inaktivieren der benutzerdefinierten Life Sciences-Funktionen“ auf Seite 507

**Zugehörige Referenzen:**

- „Benutzerdefinierte Life Sciences-Funktionen nach Funktionskategorie“ auf Seite 504

## Inaktivieren der benutzerdefinierten Life Sciences-Funktionen

Wenn Sie die benutzerdefinierten Life Sciences-Funktionen nicht mehr benötigen, können Sie die Funktionen entweder temporär inaktivieren oder permanent aus der zusammengeschlossenen Datenbank entfernen.

### Vorgehensweise:

Sie können die benutzerdefinierten Life Sciences-Funktionen mit Hilfe des Befehls **disable\_LSFfunctions** inaktivieren.

- Auf Servern mit zusammengeschlossenen Datenbanken unter Windows NT befindet sich dieser Befehl im `sql11ib\bin` Verzeichnis.
- Auf Servern mit zusammengeschlossenen Datenbanken unter AIX befindet sich dieser Befehl im Verzeichnis `sql11ib/bin`.

Die Syntax für den Befehl **disable\_LSFfunctions** lautet wie folgt:

```
disable_LSFfunctions -n dbname -u benutzer-id -p kennwort
```

### **dbname**

Der Name der zusammengeschlossenen Datenbank, für die Sie die Funktionen inaktivieren wollen.

### **benutzer-id**

Eine gültige Benutzer-ID für die zusammengeschlossene Datenbank.

### **kennwort**

Ein gültiges Kennwort für die Benutzer-ID.

### Beispiel zum Inaktivieren der benutzerdefinierten Life Sciences-Funktionen:

Das folgende Beispiel zeigt die Ausgabe, die nach Absetzen des Befehls **disable\_LSFfunctions** zurückgegeben wird:

```
C:>disable_LSFfunctions -n federateddb -u db2admin -p db2admin
```

```
(21) Life Sciences-Funktionen wurden gefunden
-- Löschen von Life Sciences-Funktionen ...
Life Sciences-Funktionen erfolgreich gelöscht.
```

\*\*\* Das Bereinigen des Systems kann einige Sekunden dauern. Bitte warten Sie .....

Um die Funktionen permanent aus der zusammengeschlossenen Datenbank zu entfernen, müssen sie deinstalliert werden.

### Zugehörige Tasks:

- „Entfernen von relationalen Wrappern, nicht relationalen Wrappern und benutzerdefinierten Life Sciences-Funktionen (Windows)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Entfernen von Wrappern, benutzerdefinierten Funktionen und Wrapper Development Kits (UNIX)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Registrieren von benutzerdefinierten Life Sciences-Funktionen“ auf Seite 505



---

## Benutzerdefinierte Rückübersetzungsfunktionen

### Benutzerdefinierte Rückübersetzungsfunktionen - Übersicht

Mit den benutzerdefinierten Rückübersetzungsfunktionen können Sie eine Peptidsequenz in eine Nukleotidsequenz umwandeln. Eine Rückübersetzung ist die Umkehrung einer Übersetzung.

Da die Zuordnung von Aminosäuren zu Nukleotid-Triplet-Codons eine Zuordnung "eins zu vielen" ist, führt eine Rückübersetzung zu zwei Ergebnissen:

#### Größte Mehrdeutigkeit

Einfache Textumwandlung und Suchfunktion. Verwenden Sie die benutzerdefinierte Funktion 'LSPep2AmbNuc', um eine Übersetzung mit der größten Mehrdeutigkeit durchzuführen.

#### Größte Wahrscheinlichkeit

Erfordert zusätzliche Informationen aus einer Codon-Frequenztafel. Verwenden Sie die benutzerdefinierte Funktion 'LSPep2ProbNuc', um eine Übersetzung mit der größten Wahrscheinlichkeit durchzuführen.

#### Zugehörige Konzepte:

- „Benutzerdefinierte Life Sciences-Funktionen - Übersicht“ auf Seite 503

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSPep2AmbNuc'“ auf Seite 508
- „Benutzerdefinierte Funktion 'LSPep2AmbNuc' - Fehlermeldungen“ auf Seite 511
- „Benutzerdefinierte Funktion 'LSPep2ProbNuc'“ auf Seite 511
- „Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Fehlermeldungen“ auf Seite 513
- „Benutzerdefinierte Funktion 'LSPep2AmbNuc' - Beispiel“ auf Seite 510
- „Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Beispiel“ auf Seite 512

### Benutzerdefinierte Funktion 'LSPep2AmbNuc'

►►—DB2LS.LSPep2AmbNuc—(*eingabe-peptidsequenz*—,dateipfad zur externen übersetzungstabelle)—►►

#### eingabe-peptidsequenz

Eine gültige Zeichenfolgedarstellung, die eine Peptidsequenz beschreibt. Eine Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 10890 Byte. Die Eingabedaten verwenden die standardmäßigen Aminosäuresymbole und Mehrdeutigkeitscodes (Ambiguitätscodes).

#### dateipfad zur externen Übersetzungstabelle

Wenn Sie eine angepasste Übersetzungstabelle verwenden, müssen Sie den entsprechenden Dateipfad angeben, damit die Tabelle gefunden wird. Der Zeichenfolgewert des Pfads darf 255 Zeichen nicht überschreiten.

Der Schemaname lautet DB2LS.

Mit Hilfe der Funktion 'LSPep2AmbNuc' erstellen Sie gemäß einer Übersetzungstabelle aus einer Peptidsequenz die Nukleotidsequenz mit der größten Mehrdeutigkeit (Ambiguität).

Das Ergebnis der Funktion ist eine Zeichenfolge vom Datentyp VARCHAR mit einer Länge von maximal 32672 Byte. Das Ergebnis stellt gemäß einer integrierten oder von Ihnen angegebenen Übersetzungstabelle die Nukleotidsequenz mit der größten Mehrdeutigkeit dar.

Wird keine Übersetzungstabelle angegeben, verwendet die Funktion standardmäßig Tabelle 96.

*Tabelle 96. Standardübersetzungstabelle*

Aminosäuresymbol	Abkürzung	Codon
A	Ala	GCX
B	Asx	RAY
C	Cys	TGY
D	Asp	GAY
V	Glu	GAR
F	Phe	TTY
G	Gly	GGX
H	His	CAY
I	Ile	ATH
K	Lys	AAR
L	Leu	YTX
M	Met	ATG
N	Asn	AAV
P	Pro	CCX
Q	Gln	CAR
R	Arg	MGX
S	Ser	WSX
T	Thr	ACX
V	Val	GTX
W	Trp	TGG
X	Xxx	XXX
Y	Tyr	TAY
Z	Glx	SAR
*	End	TRR

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSPep2AmbNuc' - Fehlermeldungen“ auf Seite 511
- „Benutzerdefinierte Funktion 'LSPep2ProbNuc'“ auf Seite 511
- „Benutzerdefinierte Funktion 'LSPep2AmbNuc' - Beispiel“ auf Seite 510

## Benutzerdefinierte Funktion 'LSPep2AmbNuc' - Beispiel

Sie können die Funktion mit einer values-Anweisung aufrufen. Die einzige Eingabe ist eine Peptidsequenz, wie in folgendem Beispiel gezeigt:

```
values db21s.LSPep2AmbNuc('HR');
```

Das vorstehende Beispiel setzt eine Peptidsequenz in eine Nukleotidsequenz um, wobei die mehrdeutigen Übersetzungen und die integrierte Übersetzungstabelle verwendet werden. Das Ergebnis der vorstehenden Anweisung ist eine Nukleotidsequenz, die aus den standardmäßigen Aminosäuresymbolen erstellt wird:

```
CAYMGX
```

Das folgende Beispiel setzt eine Peptidsequenz in eine Nukleotidsequenz um, wobei die mehrdeutigen Übersetzungen und die integrierte Tabelle verwendet werden:

```
values db21s.LSPep2AmbNuc('SRGFGFITYSHSSMIDEAQKSRPHKIDGRVVEPKRA');
```

Das Ergebnis dieser values-Anweisung ist die folgende Nukleotidsequenz (die Sequenz wurde durch Teilung an das Seitenformat angepasst):

```
WSXMGXGGXTTYGGXTTYATHACXTAYWSXCAYWSXWSXATGATHGAYGARGCXCARA  
ARWSXMGXCCXCAYAARATHGAYGGXMGXGTXTXGARCCXAARMGXGCX
```

Das nächste Beispiel zeigt die Anwendung der Funktion auf eine Gruppe von Werten, die aus einer Tabelle oder einem Kurznamen extrahiert wurden:

```
SELECT DB2LS.LsPep2AmbNuc(peptide_seq) FROM table protein_table;
```

Die Daten in der Spalte 'peptide\_seq' der Tabelle 'protein\_table' sehen folgendermaßen aus:

*Tabelle 97. Daten in der Spalte 'peptide\_seq'*

peptide_seq
GIKEDTEEHHLRDYFE
QKYHTVNGHNCEVRKA
.....

Das Ergebnis der SELECT-Anweisung sieht wie folgt aus:

```
GGXATHAARGARGAYACXGARGARCAAYCAYTTXMGXGAYTAYTTYGAR  
CARAARTAYCAYACXGTXAAYGGXCAYAAYTGYGARGTXMGXAARGCX  
...
```

Das folgende Beispiel setzt eine Peptidsequenz in eine Nukleotidsequenz um, wobei die mehrdeutigen Übersetzungen und eine benutzerdefinierte Tabelle verwendet werden. Normalerweise sind die Unterschiede zwischen den Übersetzungstabellen nur geringfügig und machen meist nur ein bis zwei eigene Symbole aus. Diese Unterschiede können vorkommen, da einige Spezies über mehr oder weniger Codons verfügen. Bei der Spezies Drosophila beispielsweise fehlt das Codon AGG.

```
values db21s.LSPep2AmbNuc('RGNMGGNYGNQGGGNWNG',  
                           '\data\transl_table_06.txt')
```

Unter der Annahme, dass die Eingabe-Übersetzungstabelle für Drosophila gilt, sieht das Ergebnis der values-Anweisung wie folgt aus:

```
MGRGGXAAAYATGGGXGGXGGXAAAYTAYGGXAAYTARAAYGGXGGXGGXAAAYTGGAAAYAYGGX
```

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSPep2AmbNuc'“ auf Seite 508
- „Benutzerdefinierte Funktion 'LSNuc2Pep' - Beispiel“ auf Seite 538

## Benutzerdefinierte Funktion 'LSPep2AmbNuc' - Fehlernachrichten

Tabelle 98. Von der benutzerdefinierten Funktion 'LSPep2AmbNuc' ausgegebene Nachrichten

Fehlercode	Nachricht	Erläuterung
SQL0443N	Die Routine "DB2LS.LSPEP2AMBNUC" (spezifischer Name "LSPEP2AMBNUC") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Die Sequenz ist ungültig". SQLSTATE=38608	Die angegebene Sequenz ist ungültig.
SQL0443N	Die Routine "DB2LS.LSPEP2AMBNUC" (spezifischer Name "LSPEP2AMBNUCUT") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Es wurde keine Übersetzung gefunden". SQLSTATE=38610	Die Datei mit der Übersetzungstabelle ist leer.
SQL0443N	Die Routine "LSPEP2AMBNUC" (spezifischer Name "LSPEP2AMBNUCUT") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Die Datei mit der Übersetzungstabelle kann nicht geöffnet werden". SQLSTATE=38612	Die angegebene Datei mit der Übersetzungstabelle existiert nicht.
SQL0443N	Die Routine "DB2LS.LSPEP2AMBNUC" (spezifischer Name "LSPEP2AMBNUCUT") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Eine Zeile war beim Lesen aus der Datei zu lang". SQLSTATE=38614	Die Datei enthielt eine Zeile, die länger war als zulässig.
SQL0443N	Die Routine "DB2LS.LSPEP2AMBNUC" (spezifischer Name "LSPEP2AMBNUCUT") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Ungültige Datendatei". SQLSTATE=38615	Das Dateiformat ist ungültig.
SQL0443N	Die Routine "LSPEP2AMBNUC" (spezifischer Name "LSPEP2AMBNUCUT") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Die Übersetzungstabelle kann nicht erstellt werden". SQLSTATE=38611	Die Datei enthielt ungültige Symbole.

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSPep2AmbNuc'“ auf Seite 508

## Benutzerdefinierte Funktion 'LSPep2ProbNuc'

►►—DB2LS.LSPep2ProbNuc—(*eingabe-peptidsequenz*—,dateipfad zur codon-frequenztafel)—►►

#### eingabe-peptidsequenz

Eine gültige Zeichenfolgedarstellung, die eine Peptidsequenz beschreibt.  
Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen

und darf nicht länger sein als 10890 Byte. Die Eingabedaten verwenden die standardmäßigen Aminosäuresymbole.

#### **dateipfad zur codon-frequenztafel**

Hierbei handelt es sich um die Codon-Frequenztafel. Geben Sie den Dateipfad an, damit die Frequenztafel gefunden wird. Der Zeichenfolgewert des Pfades darf 255 Zeichen nicht überschreiten.

Der Schemaname lautet DB2LS.

Mit der Funktion 'LSPep2ProbNuc' generieren Sie gemäß der im zweiten Argument angegebenen Codon-Frequenztafel aus einer Peptidsequenz die Nukleotidsequenz mit der größten Wahrscheinlichkeit.

Das Ergebnis der Funktion ist eine Zeichenfolge vom Datentyp VARCHAR mit einer Länge von maximal 32672 Byte. Die Zeichenfolge ist eine Darstellung der wahrscheinlichsten Nukleotidsequenz auf der Grundlage der verwendeten Codon-Frequenztafel.

#### **Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSPep2AmbNuc'“ auf Seite 508
- „Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Fehlermeldungen“ auf Seite 513
- „Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Beispiel“ auf Seite 512

## **Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Beispiel**

Das folgende Beispiel zeigt, wie Sie eine Peptidsequenz in eine Nukleotidsequenz umsetzen können, indem Sie die in der Frequenztafel 'yeast\_high.cod' definierten Übersetzungen mit der größten Wahrscheinlichkeit verwenden.

```
values db2ls.LSPep2ProbNuc('RDNDDDN', '\data\yeast_high.cod')
```

Das Ergebnis aus den vorstehenden Werten sieht wie folgt aus:

```
AGAGACAATAACGACGATGATAAC
```

Eine zweite Ausführung der gleichen Anweisung gibt die nachstehende Zeichenfolge zurück:

```
AGAGATAATAACGACGATGACAAC
```

Eine dritte Ausführung der gleichen Anweisung gibt die nachstehende Zeichenfolge mit Zufallswerten zurück:

```
AGAGATAAACAACGACGACGATAAT
```

Die fett gedruckten Codons heben jeweils die Unterschiede zwischen den aktuellen und vorherigen Umsetzungen hervor.

Die Ergebnisse aus der einzelnen values-Anweisungen zeigen, dass die Funktion 'LSPep2ProbNuc' auf Grundlage zugänglicher Statistiken eines der möglichen Symbole auswählt. Diese Funktion unterscheidet sich von Funktion 'LSPep2AmbNuc', die mehrdeutige Symbole verwendet, wenn mehrere Übersetzungen möglich sind.

Die Funktion 'LSPep2ProbNuc' berücksichtigt für jedes Symbol die Übersetzungen mit der größten Wahrscheinlichkeit, und ersetzt anschließend jedes Symbol durch eine Zufallsübersetzung aus der zuvor gewählten Gruppe. Angenommen, eine Frequenztafel enthält die folgenden Daten:

Tabelle 99. Beispieldaten einer Frequenztafel

Aminosäure	Codon	Frequenz
Ala	GCG	0.17
Ala	GCA	0.13
Ala	GCT	0.17
Ala	GCC	0.53

Angenommen, die Peptidsequenz enthält vier „A“-Symbole (Ala). Die Funktion übersetzt A zweimal in GCC, einmal in GCG und einmal in GCT. Die Reihenfolge, in der die Funktion die Übersetzungen generiert, ist jedoch willkürlich. Die Abfrage könnte das erste A in jede der Übersetzungen aus der Gruppe {GCC, GCC, GCG, GCT} umsetzen. Das Ergebnis enthält stets zwei Elemente 'GCC', ein Element 'GCG' und ein Element 'GCT' in der Ausgabe-DNA-Sequenz. Mehrere Ausführungen der Funktion für die gleiche Sequenz können unter Umständen DNA-Sequenzen zurückgeben, in denen die Werte ausgetauscht sind.

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSPep2ProbNuc'“ auf Seite 511
- „Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Fehlernachrichten“ auf Seite 513
- „Benutzerdefinierte Funktion 'LSPep2AmbNuc' - Beispiel“ auf Seite 510

## Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Fehlernachrichten

Tabelle 100. Von der benutzerdefinierten Funktion 'LSPep2ProbNuc' ausgegebene Nachrichten

Fehlercode	Nachricht	Erläuterung
SQL0443N	Die Routine "DB2LS.LSPEP2PROBNUC" (spezifischer Name "LSPEP2PROBNUC") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Die Sequenz ist ungültig". SQLSTATE=38608	Die Eingabesequenz ist ungültig.
SQL0443N	Die Routine "DB2LS.LSPEP2PROBNUC" (spezifischer Name "LSPEP2PROBNUC") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Es wurde keine Übersetzung gefunden". SQLSTATE=38610	Die Datei mit der Codon-Frequenztafel ist leer.
SQL0443N	Die Routine "LSPEP2PROBNUC" (spezifischer Name "LSPEP2PROBNUC") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Die Datei mit der Übersetzungstabelle kann nicht geöffnet werden". SQLSTATE=38612	Die Datei ist nicht vorhanden.
SQL0443N	Die Routine "DB2LS.LSPEP2PROBNUC" (spezifischer Name "LSPEP2PROBNUC") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Eine Zeile war beim Lesen aus der Datei zu lang". SQLSTATE=38614	Die Datei enthält Zeilen, die länger sind als zulässig.

Tabelle 100. Von der benutzerdefinierten Funktion 'LSPep2ProbNuc' ausgegebene Nachrichten (Forts.)

Fehlercode	Nachricht	Erläuterung
SQL0443N	Die Routine "DB2LS.LSPEP2PROBNUC" (spezifischer Name "LSPEP2PROBNUC") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Ungültige Datendatei". SQLSTATE=38615	Das Dateiformat ist ungültig.
SQL0443N	Die Routine "LSPEP2PROBNUC" (spezifischer Name "LSPEP2PROBNUC") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Die Übersetzungstabelle kann nicht erstellt werden". SQLSTATE=38611	Die Datei enthält ungültige Symbole.

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSPep2ProbNuc'“ auf Seite 511
- „Benutzerdefinierte Funktion 'LSPep2ProbNuc' - Beispiel“ auf Seite 512

## Benutzerdefinierte Funktionen für die Syntaxanalyse von Definitionszeilen

### Benutzerdefinierte Funktion für die Syntaxanalyse von Definitionszeilen - Übersicht

Die benutzerdefinierten Funktionen für die Syntaxanalyse von Definitionszeilen führen eine syntaktische Analyse von Elementen einer Definitionszeile durch. Die benutzerdefinierten Funktionen für die Syntaxanalyse von Definitionszeilen werden beispielsweise verwendet, um

- Verknüpfungen mit anderen Datenquellen gemäß Sequenzkennungen zu ermöglichen, deren syntaktische Analyse außerhalb der Definitionszeile erfolgte;
- Vergleichselemente für Abschnitte der Definitionszeile zu bewerten (beispielsweise 'species = "human"').

Die Funktionen für die Syntaxanalyse von Definitionszeilen umfassen die häufigsten Definitionszeilenformate. Zu den Beispielen gehören die Definitionszeilenelemente, die vom BLAST-Wrapper zurückgegeben werden oder die in einer Datendatei im FASTA-Format vorliegen.

**Zugehörige Konzepte:**

- „Syntaxanalyse von Definitionszeilen“ auf Seite 124

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktionen 'LSDeflineParse' - Beispiele“ auf Seite 517
- „Benutzerdefinierte LSDeflineParse-Funktionen“ auf Seite 514

### Benutzerdefinierte LSDeflineParse-Funktionen

Jede LSDeflineParse-Funktion führt eine syntaktische Analyse der Felder der NSID (NCBI Standard FASTA Sequence Identifier) und der Beschreibung durch und stellt die Ergebnisse in Spalten in einer Tabelle dar. Definitionszeilen mit zusammenge-

setzten Definitionen werden in mehreren Zeilen ausgegeben, wobei jede Zeile die Definition einer einzelnen Komponente enthält.

DB2LS ist der Schemaname, den Sie mit benutzerdefinierten Funktionen für die Syntaxanalyse von Definitionszeilen verwenden.

Es gibt die folgenden benutzerdefinierten Funktionen für die Syntaxanalyse von Definitionszeilen:

▶▶—DB2LS.LSDefineParse2—(*definitionszeile*)—————▶▶

▶▶—DB2LS.LSDefineParse3—(*definitionszeile*)—————▶▶

▶▶—DB2LS.LSDefineParse2\_2—(*definitionszeile*)—————▶▶

▶▶—DB2LS.LSDefineParse2\_3—(*definitionszeile*)—————▶▶

▶▶—DB2LS.LSDefineParse3\_3—(*definitionszeile*)—————▶▶

**definitionszeile**

Eine gültige Zeichenfolgedarstellung einer Definitionszeile im FASTA-Format. Die Zeichenfolge muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 1024 Byte.

**Die benutzerdefinierte Funktion 'LSDefineParse2':**

LSDefineParse2 führt eine syntaktische Analyse einer Definitionszeile durch, die eine aus zwei Feldern bestehende NSID enthält. Das Ergebnis der Funktion ist eine Tabelle mit vier Spalten:

*Tabelle 101. Benutzerdefinierte Funktion 'LSDefineParse2' - Spaltenbeschreibung für die Ergebnistabelle*

Spaltenname	Beschreibung
ROWID	Eine ganze Zahl zur Nummerierung der von der Funktion zurückgegebenen Zeilen.
TAG	Ein VARCHAR-Wert mit bis zu drei Zeichen zur Darstellung der NSID-Markierung.
IDENTIFIER	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des zweiten Kennungsfelds in der NSID.
DESCRIPTION	Ein VARCHAR-Wert mit bis zu 1019 Zeichen.

**Die benutzerdefinierte Funktion 'LSDefineParse3':**

LSDefineParse3 führt eine syntaktische Analyse einer Definitionszeile durch, die eine aus drei Feldern bestehende NSID enthält. Das Ergebnis der Funktion ist eine Tabelle mit fünf Spalten:



*Tabelle 102. Benutzerdefinierte Funktion 'LSDeflineParse3' - Spaltenbeschreibung für die Ergebnistabelle*

Spaltenname	Beschreibung
ROWID	Eine ganze Zahl zur Nummerierung der von der Funktion zurückgegebenen Zeilen.
TAG	Ein VARCHAR-Wert mit bis zu drei Zeichen zur Darstellung der NSID-Markierung.
ACCESSION	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des zweiten Kennungsfelds in der NSID.
LOCUS	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des dritten Kennungsfelds in der NSID.
DESCRIPTION	Ein VARCHAR-Wert mit bis zu 1017 Zeichen.

#### **Die benutzerdefinierte Funktion 'LSDeflineParse2\_2':**

LSDeflineParse2\_2 führt eine syntaktische Analyse einer Definitionszeile durch, die über eine zusammengesetzte Kennung verfügt, die aus einem Paar von zwei verknüpften NSIDs mit jeweils zwei Feldern besteht. Das Ergebnis der Funktion ist eine Tabelle mit sechs Spalten:

*Tabelle 103. Benutzerdefinierte Funktion 'LSDeflineParse2\_2' - Spaltenbeschreibung für die Ergebnistabelle*

Spaltenname	Beschreibung
ROWID	Eine ganze Zahl zur Nummerierung der von der Funktion zurückgegebenen Zeilen.
TAG1	Ein VARCHAR-Wert mit bis zu drei Zeichen zur Darstellung der NSID-Markierung der ersten Kennung.
IDENTIFIER1	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des zweiten Kennungsfelds der ersten NSID.
TAG2	Ein VARCHAR-Wert mit bis zu drei Zeichen zur Darstellung der NSID-Markierung der zweiten Kennung.
IDENTIFIER2	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des zweiten Kennungsfelds der zweiten NSID.
DESCRIPTION	Ein VARCHAR-Wert mit bis zu 1015 Zeichen.

#### **Die benutzerdefinierte Funktion 'LSDeflineParse2\_3':**

LSDeflineParse2\_3 führt eine syntaktische Analyse einer Definitionszeile durch, die über eine zusammengesetzte Kennung verfügt, die aus einer NSID mit zwei Feldern besteht, die mit einer aus drei Feldern bestehenden NSID verknüpft ist. Die Reihenfolge der Verknüpfung in der Eingabedefinitionszeile ist unerheblich, das heißt, es spielt keine Rolle, ob die NSID mit zwei Feldern vor der NSID mit drei Feldern steht oder umgekehrt. Das Ergebnis der Funktion ist eine Tabelle mit sieben Spalten:

*Tabelle 104. Benutzerdefinierte Funktion 'LSDeflineParse2\_3' - Spaltenbeschreibung für die Ergebnistabelle*

Spaltenname	Beschreibung
ROWID	Eine ganze Zahl zur Nummerierung der von der Funktion zurückgegebenen Zeilen.

*Tabelle 104. Benutzerdefinierte Funktion 'LSDeflineParse2\_3' - Spaltenbeschreibung für die Ergebnistabelle (Forts.)*

<b>Spaltenname</b>	<b>Beschreibung</b>
TAG1	Ein VARCHAR-Wert mit bis zu drei Zeichen zur Darstellung der NSID-Markierung der aus zwei Feldern bestehenden Kennung.
IDENTIFIER	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des zweiten Kennungsfelds der aus zwei Feldern bestehenden NSID.
TAG2	Ein VARCHAR-Wert mit bis zu drei Zeichen zur Darstellung der NSID-Markierung der aus drei Feldern bestehenden Kennung.
ACCESSION	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des zweiten Kennungsfelds der aus drei Feldern bestehenden NSID.
LOCUS	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des dritten Kennungsfelds der aus drei Feldern bestehenden NSID.
DESCRIPTION	Ein VARCHAR-Wert mit bis zu 1013 Zeichen.

### **Die benutzerdefinierte Funktion 'LSDeflineParse3\_3':**

LSDeflineParse3\_3 führt eine syntaktische Analyse einer Definitionszeile durch, die über eine zusammengesetzte Kennung verfügt, die aus einem Paar von NSIDs mit jeweils drei Feldern besteht. Das Ergebnis der Funktion ist eine Tabelle mit acht Spalten:

*Tabelle 105. Benutzerdefinierte Funktion 'LSDeflineParse3\_3' - Spaltenbeschreibung für die Ergebnistabelle*

<b>Spaltenname</b>	<b>Beschreibung</b>
ROWID	Eine ganze Zahl zur Nummerierung der von der Funktion zurückgegebenen Zeilen.
TAG1	Ein VARCHAR-Wert mit bis zu drei Zeichen zur Darstellung der NSID-Markierung der ersten Kennung.
ACCESSION1	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des zweiten Kennungsfelds der ersten NSID.
LOCUS1	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des dritten Kennungsfelds der ersten NSID.
TAG2	Ein VARCHAR-Wert mit bis zu drei Zeichen zur Darstellung der NSID-Markierung der ersten Kennung.
ACCESSION2	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des zweiten Kennungsfelds der zweiten NSID.
LOCUS2	Ein VARCHAR-Wert mit bis zu 20 Zeichen zur Darstellung des dritten Kennungsfelds der zweiten NSID.
DESCRIPTION	Ein VARCHAR-Wert mit bis zu 1014 Zeichen.

### **Zugehörige Referenzen:**

- „Benutzerdefinierte Funktionen 'LSDeflineParse' - Beispiele“ auf Seite 517

## **Benutzerdefinierte Funktionen 'LSDeflineParse' - Beispiele**

Dieser Abschnitt enthält sieben Beispiele, die zeigen, wie die benutzerdefinierten LSDeflineParse-Funktionen aus Definitionszeilen mittels syntaktischer Analyse Ergebnistabellen erstellen.

Die folgende Beispielabfrage und Ergebnistabelle zeigen, wie die benutzerdefinierte Funktion 'LSDeflineParse2' eine Definitionszeile, die eine aus zwei Feldern bestehende NSID enthält, syntaktisch analysiert:

```
select *
from table(DB2LS.LSDeflineParse2(
    '>gi|12346 hypothetical protein 185 -wheat chloroplast')) as t
```

Die Ergebnistabelle enthält die folgenden Daten:

*Tabelle 106. Ergebnisdaten der benutzerdefinierten Funktion 'LSDeflineParse2'*

Spaltenname	Daten
ROWID	1
TAG	gi
IDENTIFIER	12346
DESCRIPTION	hypothetical protein 185 – wheat chloroplast

Die folgende Beispielabfrage und Ergebnistabelle zeigen, wie die benutzerdefinierte Funktion 'LSDeflineParse3' eine Definitionszeile, die eine aus drei Feldern bestehende NSID enthält, syntaktisch analysiert:

```
select *
from table(DB2LS.LSDeflineParse3(
    '>gb|U37104|APU37104 Aethia pusilla cytochrome b gene')) as t
```

Die Ergebnistabelle enthält die folgenden Daten:

*Tabelle 107. Ergebnisdaten der benutzerdefinierten Funktion 'LSDeflineParse3'*

Spaltenname	Daten
ROWID	1
TAG	gb
ACCESSION	U37104
LOCUS	APU37104
DESCRIPTION	Aethia pusilla cytochrome b gene

Die folgende Beispielabfrage und Ergebnistabelle zeigen, wie die benutzerdefinierte Funktion 'LSDeflineParse2\_2' eine Definitionszeile syntaktisch analysiert, die eine zusammengesetzte Kennung enthält, die aus einem Paar aus NSIDs mit jeweils zwei Feldern besteht:

```
select *
from table(DB2LS.LSDeflineParse2_2(
    '>gb|U37104|gim|73401A Aethia pusilla cytochrome b gene')) as t
```

Die Ergebnistabelle enthält die folgenden Daten:

*Tabelle 108. Ergebnisdaten der benutzerdefinierten Funktion 'LSDeflineParse2\_2'*

Spaltenname	Daten
ROWID	1
TAG1	gb
IDENTIFIER1	U37104
TAG2	gim
IDENTIFIER2	73401A

*Tabelle 108. Ergebnisdaten der benutzerdefinierten Funktion 'LSDefineParse2\_2' (Forts.)*

Spaltenname	Daten
DESCRIPTION	Aethia pusilla cytochrome b gene

Die folgende Beispielabfrage enthält eine Definitionszeile mit einer zusammengesetzten Kennung, die sich aus einer NSID mit 2 Feldern zusammensetzt, die mit einer aus drei Feldern bestehenden NSID verknüpft ist. Das Beispiel zeigt, wie die Funktion 'LSDefineParse2\_3' diese Definitionszeile syntaktisch analysiert:

```
select *
from table(DB2LS.LSDefineParse2_3('
>gi|12346|gp|CAA44030.1|CHTAHSRA_4
hypothetical protein 185 - wheat chloroplast')) as t
```

Die Ergebnistabelle enthält die folgenden Daten:

*Tabelle 109. Ergebnisdaten der benutzerdefinierten Funktion 'LSDefineParse2\_3'*

Spaltenname	Daten
ROWID	1
TAG1	gi
IDENTIFIER	12346
TAG2	gp
ACCESSION	CAA44030.1
LOCUS	CHTAHSRA_4
DESCRIPTION	hypothetical protein 185 - wheat chloroplast

Die folgende Beispielabfrage enthält eine Definitionszeile mit einer zusammengesetzten Kennung, die sich aus einer NSID mit 3 Feldern zusammensetzt, die mit einer aus zwei Feldern bestehenden NSID verknüpft ist. Das Beispiel zeigt, wie die Funktion 'LSDefineParse2\_3' diese Definitionszeile syntaktisch analysiert:

```
select *
from table(DB2LS.LSDefineParse2_3('
>gp|CAA44030.1|CHTAHSRA_4|gi|12346
hypothetical protein 185 - wheat chloroplast')) as t
```

Die Ergebnistabelle enthält die folgenden Daten:

*Tabelle 110. Ergebnisdaten der benutzerdefinierten Funktion 'LSDefineParse2\_3'*

Spaltenname	Daten
ROWID	1
TAG1	gi
IDENTIFIER	12346
TAG2	gp
ACCESSION	CAA44030.1
LOCUS	CHTAHSRA_4
DESCRIPTION	hypothetical protein 185 - wheat chloroplast

Die folgende Beispielabfrage und Ergebnistabelle zeigen, wie die benutzerdefinierte Funktion 'LSDeflineParse3\_3' eine Definitionszeile syntaktisch analysiert, die eine zusammengesetzte Kennung enthält, die aus einem Paar aus NSIDs mit jeweils drei Feldern besteht:

```
select * from table(DB2LS.LSDeflineParse3_3('
>dbj|AAD55586.1|AF055084_1|gp|CAA44030.1|CHTAHSRA_4
hypothetical protein 185 - wheat chloroplast')) as t
```

Die Ergebnistabelle enthält die folgenden Daten:

*Tabelle 111. Ergebnisdaten der benutzerdefinierten Funktion 'LSDeflineParse3\_3'*

Spaltenname	Daten
ROWID	1
TAG1	dbj
ACCESSION1	AAD55586.1
LOCUS1	AF055084_1
TAG2	gp
ACCESSION2	CAA44030.1
LOCUS2	CHTAHSRA_4
DESCRIPTION	hypothetical protein 185 - wheat chloroplast

Zur syntaktischen Analyse einer zusammengesetzten Definitionszeile kann eine beliebige der benutzerdefinierten defline-Funktionen verwendet werden. Die folgende Beispielabfrage enthält eine zusammengesetzte Definitionszeile mit mehreren Definitionen, die durch ein Steuerung-A-Zeichen voneinander getrennt werden. Dieser Definitionstyp kommt in der NCBI-Datenbank 'nr' der nicht-redundanten Proteine vor. Das Beispiel zeigt, wie die Funktion 'LSDeflineParse2\_3' diese Definitionszeile syntaktisch analysiert:

```
select *
from table(DB2LS.LSDeflineParse2_3('
>gi|12346|gp|CAA44030.1|CHTAHSRA_4
hypothetical protein 185 - wheat chloroplast
^Agp|CAA44030.1|CHTAHSRA_4|gi|12346
hypothetical protein 185 - wheat chloroplast')) as t
```

Die Ergebnistabelle enthält die folgenden Daten:

*Tabelle 112. Ergebnisdaten der benutzerdefinierten Funktion 'LSDeflineParse2\_3'*

Spaltenname	Daten	Daten
ROWID	1	2
TAG1	gi	gi
IDENTIFIER	12346	12346
TAG2	gp	gp
ACCESSION	CAA44030.1	CAA44030.1
LOCUS	CHTAHSRA_4	CHTAHSRA_4
DESCRIPTION	hypothetical protein 185 - wheat chloroplast	hypothetical protein 185 - wheat chloroplast

#### Zugehörige Referenzen:

- „Benutzerdefinierte LSDeflineParse-Funktionen“ auf Seite 514

---

## Benutzerdefinierte Funktionen für allgemeine Mustererkennung

Die benutzerdefinierten Funktionen für allgemeine Mustererkennung identifizieren relevante Bereiche in einer angegebenen Zeichenfolge, wie beispielsweise Nukleotid- oder Peptidsequenzen.

### Benutzerdefinierte Funktion 'LSPatternMatch'

►►—DB2LS.LSPatternMatch—(eingabe-zeichenfolge, muster)—◀◀

#### eingabe-zeichenfolge

Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

#### muster

Das Muster gemäß Angabe in einem gültigen regulären PERL-Ausdruck. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

Der Schemaname lautet DB2LS.

Mit Hilfe der benutzerdefinierten Funktion 'LSPatternMatch' können Sie die Eingabe-Nukleotidsequenz oder Eingabe-Peptidsequenz nach einem angegebenen Muster durchsuchen.

Das Ergebnis der Funktion ist eine ganze Zahl zur Darstellung der Position der ersten Übereinstimmung des Musters in der Sequenz. Ist keine Übereinstimmung vorhanden, gibt die Funktion den Wert Null zurück.

Sind Muster vorhanden, die in der PROSITE-Syntax geschrieben wurden, können Sie diese mit Hilfe der benutzerdefinierten Funktion 'LSPrositePattern' in die PERL-Syntax umwandeln. Anschließend können Sie die umgewandelte Syntax mit der benutzerdefinierten Funktion 'LSPatternMatch' verwenden.

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSPatternMatch' - Beispiel“ auf Seite 521
- „Benutzerdefinierte Funktion 'LSPrositePattern'“ auf Seite 523

### Benutzerdefinierte Funktion 'LSPatternMatch' - Beispiel

In folgendem Beispiel wird nach der Anfangsposition der Zeichenfolge gesucht, die „coward“, „cowage“, „cowboy“ oder „cowl“ entspricht.

```
values DB2LS.LSPatternMatch('joe the cowboy is next', 'cow(ard|age|boy|l)')
```

Die Funktion sucht nach Zeichen und gibt in diesem Beispiel den Wert neun zurück. Die Zeichenfolge „cowboy“ beginnt an Position neun, wobei davon ausgegangen wird, dass die erste Position eins ist.

Im nächsten Beispiel wird nach der Anfangsposition der Zeichenfolge gesucht, die „not “ oder „non“ entspricht:

```
values DB2LS.LSPatternMatch('match not and non but  
no match for no or none', 'no[tn]')
```

Die Funktion sucht nach Zeichen und gibt in diesem Beispiel den Wert sieben zurück. Die Zeichenfolge „not“ beginnt an Position sieben, wobei davon ausgegangen wird, dass die erste Position eins ist.

'LSPatternMatch' ist nützlich in SELECT-Anweisungen zum Filtern der Ergebnisse anhand der PERL-Syntax. Diese Syntax ist leistungsfähiger als die SQL-Anweisung LIKE. In folgendem Beispiel wird die Funktion 'LSPatternMatch' für eine BLAST-Ausgabe verwendet, um die Gene zu filtern, die einem bestimmten Muster entsprechen:

```
SELECT BlastOutput.*
FROM BlastOutput
WHERE db21s.LSPatternMatch(HSP_H_Seq, 'F[GSTV]PRL') > 0;
```

Wenn Sie mit der PROSITE-Syntax besser vertraut sind, können Sie mit der vorstehenden Abfrage die Funktion 'LSPrositePattern' verwenden. Ändern Sie in diesem Fall die Abfrage wie folgt:

```
SELECT BlastOutput.*
FROM BlastOutput
WHERE db21s.LSPatternMatch(HSP_H_Seq,
db21s.LSPrositePattern('F-[GSTV]-P-R-L.')) > 0;
```

Die Mustererkennungsfunktionen sind nützlich für die Suche in anderen Texttypen sowie in Nukleotid- und Peptidsequenzen. Falls die Leistungsfähigkeit ein Problem darstellen könnte, sollte die Verwendung der SQL-Anweisung LIKE in Betracht gezogen werden.

Das folgende Beispiel zeigt eine Abfrage, die BLAST-HSP-Alignments auf der Grundlage von Proteinmotiven filtert, die in der Betreff- oder Zielzeile des Alignments gefunden wurden. Grundlage des Beispiels ist folgende Publikation: Zhang, Z., Schaffer, A.A., Miller, W., Madden, T.L., Lipman, D.J., Koonin, E.V. and Altschul, S.F. (1998) Protein sequence similarity searches using patterns as seeds. *Nucl. Acids Res.*, **26**, 3896-3990.

Die folgende Abfrage gibt nur diejenigen Alignments zurück, in denen die betreffende Sequenz die P-Schleife-ATPase-Domäne [GA]xxxxGK[ST] enthält. Die Abfrage verwendet das Protein CED4 (den Apoptoseregulator des Nematoden *Caenorhabditis elegans*) als Abfragesequenz für die NCBI-Datenbank der nicht-redundanten Proteinsequenzen. Die Datenbank ruft die BLAST-Abfragesequenz aus der Übersetzung des CDS-Merkmals des GenBank-Eintrags X69016 ab.

```
SELECT HSP_Q_Seq, HSP_Midline, HSP_H_Seq
FROM BlastP b, GBseq gs, gbfeat gf, gbqual gq
WHERE gs.PRIMARYACCESSION = 'X69016' and
      gs.sequencekey = gf.sequencekey and
      gf.featurejoinkey = gq.featurejoinkey and
      gf.FeatureKey = 'CDS' and
      gq.QualifierName = 'translation' and
      gq.QualifierValue = b.BlastSeq and
      db21s.LSPatternMatch(HSP_H_Seq,
db21s.LSPrositePattern('[GA]-x(4)-G-K-[ST].')) > 0;
```

Mit Hilfe der nächsten Beispielabfrage können Sie in einer Genomsequenz nach HSPs suchen, die mutmaßliche Einzel-Nukleotid-Polymorphismen (Single Nucleotide Polymorphisms, SNPs) im Hinblick auf eine kanonische Abfragesequenz enthalten. Grundlage dieser Abfrage ist folgende Publikation: Extending traditional query-based integration approaches for functional characterization of post-genomic data. (2001) Barbara A Eckman, Anthony S Kosky, and Leonardo A Laroco Jr. *Bioinformatics* 17(7), 587-601.

Die Abfrage verwendet die Mustererkennung für die BLAST-HSP-Mittellinie (Midline), um ein Muster von  $\geq 20$  genauen Übereinstimmungen zu finden, gefolgt von einer einzelnen Diskrepanz, gefolgt von  $\geq 20$  genauen Übereinstimmungen, das heißt, zunächst 20 "|" -Zeichen, dann ein einzelnes Leerzeichen und anschließend 20 "|" -Zeichen in der Mittellinie des Alignments.

Außerdem zeigt dieses Beispiel die Verwendung der benutzerdefinierten Funktion 'LSPatternMatch' für Zeichenfolgen, bei denen es sich nicht um Nukleotid- oder Peptidsequenzen handelt.

```
SELECT HSP_Info, HSP_Midline, HSP_H_Seq
FROM BlastOutput
WHERE db21s.LSPatternMatch(HSP_Midline, '\|{20} \|{20}') > 0;
```

Sie können die vorstehende Abfrage wie folgt umschreiben:

```
SELECT HSP_Info, HSP_Midline, HSP_H_Seq, func.Position, func.Match
FROM BlastOutput,
TABLE(SELECT * AS c FROM TABLE(
LSMultiMatch(HSP_Midline, '\|{20} \|{20}') )
AS f) AS func
```

Diese zweite Abfrage gibt diejenigen BLAST-Zeilen zurück, die eine Übereinstimmung enthalten, zusammen mit der passenden Zeichenfolge und den entsprechenden Positionen in der Sequenz.

Die BLAST-Ausgabe (BlastOutput) ist eine Sicht eines BlastN-Kurznamens.

**Zugehörige Referenzen:**

## Benutzerdefinierte Funktion 'LSPrositePattern'

►—DB2LS.LSPrositePattern—(*muster*)—►

### **muster**

Das Muster, das der von der PROSITE-Syntax angegebenen Syntax entspricht. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

Der Schemaname lautet DB2LS.

Mit Hilfe der benutzerdefinierten Funktion 'LSPrositePattern' können Sie eine Umwandlung von der PROSITE-Syntax in die PERL-Syntax vornehmen. Anschließend können Sie die umgewandelte Syntax mit den benutzerdefinierten Funktionen 'LSPatternMatch', 'LSMultiMatch' und 'LSMultiMatch3' verwenden.

Das Ergebnis der Funktion ist eine Zeichenfolge zur Darstellung eines regulären Ausdrucks in der PERL-Syntax. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSPrositePattern' - Beispiel“ auf Seite 524
- „Benutzerdefinierte Funktion 'LSPatternMatch'“ auf Seite 521



## Benutzerdefinierte Funktion 'LSPrositePattern' - Beispiel

In folgendem Beispiel wird ein Muster aus der PROSITE-Syntax in die PERL-Syntax umgewandelt.

```
values db21s.LSPrositePattern('[AC]-x-V-x(4)-{ED}');
```

Die Funktion wandelt das Eingabemuster in PROSITE-Syntax in ein äquivalentes Muster in PERL-Syntax um, wie aus folgendem Beispiel ersichtlich wird:

```
[AC].V.{4}[^ED]
```

Im nächsten Beispiel wird ein anderes Muster aus der PROSITE-Syntax in die PERL-Syntax umgewandelt:

```
values db21s.LSPrositePattern('<A-x-[ST](2)-x(0,1)-V.');
```

Die Funktion übersetzt die Zeichenfolge aus der PROSITE-Syntax auf Grundlage des Eingabemusters und gibt Folgendes zurück:

```
\AA.[ST]{2}.{0,1}V
```

Im nächsten Beispiel wird das Muster, das dem PROSITE-Datenbankeintrag mit der ID-Nummer PS01205 entspricht, in ein PERL-Muster umgewandelt, das von den Mustererkennungsfunktionen als Eingabe verwendet wird.

```
values db21s.LSPrositePattern('R-P-L-[IV]-x-[NS]-F-G-S-[CA]-T-C-P-x-F.');
```

Die Abfrage gibt folgendes Ergebnis zurück:

```
RPL[IV].[NS]FGS[CA]TCP.F
```

Das nächste Beispiel zeigt, wie Sie die Funktion in einer Abfrage verwenden können. Die Abfrage druckt nur diejenigen Sequenzen aus, die dem angegebenen PROSITE-Muster entsprechen.

```
SELECT H_Accession, HSP_Info, HSP_H_Seq
FROM BlastOutput
WHERE db21s.LSPatternMatch( HSP_H_Seq,
  db21s.LSPrositePattern('R-P-L-[IV]-x-[NS]-F-G-S-[CA]-T-C-P-x-F.')
```

Im nächsten Beispiel wird das Muster umgewandelt, das dem PROSITE-Eintrag mit der ID PS00261 entspricht:

```
values db21s.LSPrositePattern('C-[STAGM]-G-[HFYL]-C-x-[ST].');
```

Die Abfrage gibt folgendes Ergebnis zurück:

```
C[STAGM]G[HFYL]C.[ST]
```

## Unterstützung für reguläre Ausdrücke

Die Unterstützung für reguläre Ausdrücke wird vom PCRE-Bibliothekspaket zur Verfügung gestellt, einer von Philip Hazel geschriebenen Software mit offener Quelle, deren Urheberrecht bei der University of Cambridge, England, liegt.

Die Quelle befindet sich unter  
<ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/>.

#### Zugehörige Konzepte:

- „Benutzerdefinierte Life Sciences-Funktionen - Übersicht“ auf Seite 503

---

## Benutzerdefinierte Funktion 'GeneWise'

Die benutzerdefinierte Funktion 'GeneWise' führt ein Alignment einer Proteinsequenz mit einer Genomsequenz durch.

'GeneWise' ist eine häufig verwendete Komponente, die ein Alignment einer Proteinsequenz mit einer genomischen DNA-Sequenz durchführt, unter Berücksichtigung von Introns- und Frameshifting-Fehlern.

### Verbindung zu 'GeneWise' herstellen

Dieser Abschnitt beschreibt, wie eine Verbindung zur Bibliothek 'GeneWise' hergestellt wird.

#### Vorgehensweise:

Um eine Verbindung zur Bibliothek 'GeneWise' herzustellen, gehen Sie wie folgt vor:

1. Laden Sie das Wise2-Paket Version 2.1.20c von '[www.ebi.ac.uk/Wise2](http://www.ebi.ac.uk/Wise2)' herunter.
2. Dekomprimieren Sie das Archiv im gewünschten Ordner.
3. Kompilieren Sie das Paket mit pthread-Unterstützung. Die Dokumentation von Wise2 enthält weitere Informationen zu diesem Schritt.

Auf HP-UX-Servern mit zusammengeschlossenen Datenbanken: Sie müssen die Option `+z` zu den Compileroptionen hinzufügen, bevor Sie den Quellcode kompilieren. Um diese Option hinzuzufügen, öffnen Sie die Datei `makefile` über das Stammverzeichnis des Wise2-Pakets und ändern Sie die Zeile wie in nachfolgendem Beispiel gezeigt:

```
CFLAGS = -c -O -DPTHREAD to: CFLAGS = -c -O -DPTHREAD +z
```

4. Führen Sie den Befehl **make api** im entsprechenden Stammverzeichnis aus.
5. Setzen Sie die Umgebungsvariable `WISE2_HOME` so, dass sie auf das Stammverzeichnis des Wise2-Pakets verweist.
6. Setzen Sie die Variable `WISECONFIGDIR` in der Datei `sqllib/cfg/db2dj.ini` so, dass sie auf das Unterverzeichnis 'wisecfg' verweist. Beispiel: Ist das Wise2-Paket im Verzeichnis `/usr/wise2.1.20c/` installiert, fügen Sie der Datei `db2dj.ini` Folgendes hinzu: `WISECONFIGDIR=/usr/wise2.1.20c/wisecfg/`
7. Führen Sie die Prozedur **djxlinkLSGeneWise** aus, die sich im Verzeichnis `sqllib/bin` befindet.
8. Überprüfen Sie die Ausgabe der Prozedur **djxlinkLSGeneWise**. Diese Ausgabe-datei (`djxlinkLSGeneWise.out`) befindet sich im Verzeichnis `sqllib/function`.
9. Wurden keine Fehler dokumentiert, wurde die Bibliothek erfolgreich erstellt.

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSGeneWise'“ auf Seite 526

## Benutzerdefinierte Funktion 'LSGeneWise'

►►—DB2LS.LSGeneWise—(proteinsequenz, DNA-sequenz)—►►

### proteinsequenz

Eine gültige Zeichenfolgedarstellung, die eine Peptidsequenz beschreibt. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

### DNA-sequenz

Eine gültige Zeichenfolgedarstellung, die eine Nukleotidsequenz beschreibt. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

Tabelle 113 zeigt die von der Funktion 'LSGeneWise' zurückgegebene einzeilige Ausgabetablelle

Tabelle 113. Spaltennamen, Typen und Beschreibungen für die von der Funktion 'LSGeneWise' zurückgegebene Ausgabetablelle

Spaltenname	Typ	Beschreibung
PROTEIN_OFFSET	INTEGER	Zeigt das Start-Offset in der Eingabe-Proteinsequenz, an dem ein Alignment gefunden wurde.
DNA_OFFSET	INTEGER	Zeigt das Start-Offset in der Eingabe-DNA-Sequenz, an dem ein Alignment gefunden wurde.
PROTEIN	VARCHAR(32672)	Ein Fragment aus der Eingabesequenz, das die alignierte Sequenz darstellt.
SIMILARITY	VARCHAR(32672)	Zeigt die Übereinstimmung zwischen dem Protein und der DNA-Sequenz. Perfekte Übereinstimmungen werden mit dem entsprechenden Symbolbuchstaben markiert. Nicht perfekte Übereinstimmungen mit einem positiven Score werden mit einem Pluszeichen ("+") und Diskrepanzen mit einem Leerzeichen markiert.
TRANSLATED_DNA	VARCHAR(32672)	Die übersetzte DNA-Sequenz. Unter Umständen enthält die Sequenz Gedankenstriche und Sondersymbole wie beispielsweise Deletionen und Introns.
DNA	VARCHAR(32672)	Die DNA-Sequenz mit Sondermarkierungen wie beispielsweise Frameshifting und Introns.

Die Übereinstimmung zwischen der Ausgabe des Programms 'GeneWise' und der Ausgabe der benutzerdefinierten Funktion 'LSGeneWise' sieht wie folgt aus:

- Die vom Programm 'GeneWise' gedruckten Protein- und DNA-Offsets entsprechen den Spalten PROTEIN\_OFFSET und DNA\_OFFSET.
- Die von 'GeneWise' in der ersten Zeile gedruckte Proteinsequenz entspricht der Spalte PROTEIN.
- Die Zeile 'similarity', die zweite Zeile in der Ausgabe von 'GeneWise', entspricht der Spalte SIMILARITY.
- Die dritte Zeile in der Ausgabe von 'GeneWise' entspricht der Spalte TRANSLATED\_DNA.

- Die vierte, fünfte und sechste Zeile der Ausgabe von 'GeneWise' sind (vertikal gelesen) zur Spalte DNA zusammengefasst.

Mit der benutzerdefinierten Funktion 'LSGeneWise' können Sie ein Alignment einer Proteinsequenz mit einer genomischen DNA-Sequenz durchführen, unter Berücksichtigung von Introns- und Frameshifting-Fehlern.

Weitere Informationen zur Ausgabe der benutzerdefinierten Funktion 'LSGeneWise' finden Sie unter 'http://www.ebi.ac.uk/Wise2'.

**Zugehörige Tasks:**

- „Verbindung zu 'GeneWise' herstellen“ auf Seite 525

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSGeneWise' - Beispiel“ auf Seite 527

## Benutzerdefinierte Funktion 'LSGeneWise' - Beispiel

Das folgende Beispiel zeigt eine Abfrage mit der benutzerdefinierten Funktion 'LSGeneWise' und die entsprechenden Ergebnisdaten.

```
select protein_offset, dna_offset, protein, similarity, translated_dna, dna
from table( db21s.LSGeneWise( '
VEPKRAVPRQIDSPNAGATVKKLFVVGALKDDHDEQSIRDYFQHFQGNIVDINIVIDKETGK
KRGFAFVEFDDYDPVDKVVVLQKQHQLNGKMVDVKKALPKQNDQQGGGGRRGGPGRAGGNR
GNMGGGNYGNQGGGNWNNGGNNWGNR' ,
'CACTTAACTGTGAAAGATATTTGGTGGCATTAAAGAAGACACTGAAGAACATCACCTAAG
AGATTATTTGAACAGTATGGAAAATTGAAGTGATTGAAATCATGACTGACCGAGGCAGTGG
CAAGAAAAGGGGCTTTGCCTTRGTAACCTTTGACGACCATGACTCCGTGGATAAGATTGTCAT
TCAGAAATACCATACTGTGAATGGCCACAACCTGTGAAGTTAGAAAAGCCCTGTCAAAGCAAGA
GATGGCTAGTGCTTCATCCAGCCAAAGAGTCAAGTGGTTCTGAAAACCTTTGGTGGTGGTTCG
TGGAGGTGGTTTCGGTGGGAATGACAACCTTCGGTCTGTGGAGGAAACTTCAGTGGTCTGGTYG
CTTTGGTGGCAGCCGTGGTGGTGGATATGGTGGC' ) ) as f;
```

Tabelle 114. Ergebnistabelle

Spalte	Daten
PROTEIN_OFFSET	23
DNA_OFFSET	14
PROTEIN	KLVFGALKDDHDEQSIRDYFQHFQGNIVDINIVIDKETG GKKRGRFAFVEFDDYDPVDKVVVLQKQHQLNGKMVD VKKALPKQNDQQGGGGRRGGPGRAGGNRGNMGG GNYGNQNGGGNWNNGGN
SIMILARITY	K+FVG +K+D +E +RDYF+ +G I I I+ D+ +GKKRGRFA+V FDD+D VDK+V+QK H +NG +V+KAL KQ RG G GN+GGG G G N+ GGN
TRANSLATED_DNA	KIFVGGIKEDTEEHHLRDYFEQYQKIEVIEIMTDRGSGK KRGFAxVTFDDHDSVDKIVIQKYHTVNGHNCEVRKAL SKQEMASASSQRGRSGS----- GNFGGGRGGGFGGNDNFGRGGN
DNA	aagatatttgggtggcattaagaagacactgaagaacatcacctaagagat...

**Zugehörige Tasks:**

- „Verbindung zu 'GeneWise' herstellen“ auf Seite 525

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSGeneWise'“ auf Seite 526

---

## Benutzerdefinierte Motivfunktionen

Benutzerdefinierte Motivfunktionen suchen übereinstimmende Muster in Nukleotid- oder Aminosäuresequenzen.

### Benutzerdefinierte Funktion 'LSBarCode'

►—DB2LS.LSBarCode—(eingabe-zeichenfolge)—◄

#### eingabe-zeichenfolge

Eine gültige Zeichenfolge zur Darstellung einer HSP-Alignments zwischen zwei Sequenzfragmenten. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

Der Schemaname lautet DB2LS.

Mit der benutzerdefinierten Funktion 'LSBarCode' wird anhand einer Eingabesequenz eine andere Sequenz generiert, indem jedes Zeichen (außer Leerzeichen und Pluszeichen) durch ein vertikales Balkensymbol (|) ersetzt wird.

Das Ergebnis der Funktion ist eine variable Zeichenfolge zur Darstellung einer Barcode-Sequenz.

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSBarCode' - Beispiel“ auf Seite 528
- „Benutzerdefinierte Funktion 'LSMultiMatch'“ auf Seite 529
- „Benutzerdefinierte Funktion 'LSMultiMatch3'“ auf Seite 531

### Benutzerdefinierte Funktion 'LSBarCode' - Beispiel

In diesem Beispiel wird aus einer Zeichenfolgesequenz ein Barcode erstellt:

```
values db2ls.LSBarCode(
  'MDY +G++L GN ++ +PASLTK+MT YVV +A+ + +I D+VTVG+DAWA NP ')
```

Das Ergebnis aus den vorstehenden Werten sieht wie folgt aus:

```
||| +|++| || ++ +|||||+|| ||| +|+ + +| |+||||+|||| ||
```

Das nächste Beispiel zeigt eine realistischere Verwendung dieser Funktion. Angenommen, ein Forscher führt eine BLAST-Suche durch und will als Rückgabewerte nur solche HSP-Alignments erhalten, die unter ihren genauen Übereinstimmungen einen Anteil an Prolinen von weniger als 25% haben. In diesem Beispiel wird mit Hilfe der Funktion der Prozentsatz an Prolinen (Symbol 'P') unter den genauen Übereinstimmungen in einem von BLAST zurückgegebenen Alignment berechnet. Hierbei ist zu beachten, dass in diesem Beispiel auch die benutzerdefinierte Funktion 'LSMultiMatch3' aufgerufen wird. Die Abfrage verwendet eine Abgleichsfunktion, um genaue Übereinstimmungen zu ermitteln. Sie wird in dieser Abfrage zusammen mit der Funktion 'LSBarCode' verwendet, da BLAST in einem Alignment nicht immer eine Folge von Balken („|“) zurückgibt. Das nachstehende Beispiel zeigt dies:

```
Query: MDYTTGQILTAGNEHQQRNPASLTKLMTGYVVVDRAIDSHRITPDDIVTVGRDAWAKDNPV
Alignment: MDY +G++L GN ++ +PASLTK+MT YVV +A+ + +I D+VTVG+DAWA NP
Target: MDYASGKVLAEGRNADEKLDPASLTKIMTSYVVGQALKADKIKLTDMMVTGKDAWATGNPA
```

Verwenden Sie die Funktion 'LSBarCode', um zu gewährleisten, dass die Ausgabe mit der korrekten Balkenfolge ausgerichtet (aligniert) wird. Die Funktion ersetzt alle Zeichen (ausgenommen Leerzeichen und Pluszeichen) durch einen vertikalen Balken.

```
SELECT BlastOutput.* , float( p )/ float( m ) AS percent_prolines
FROM BlastOutput b,
  table(SELECT COUNT(*) AS p FROM table(
    db21s.LSMultiMatch3(
      b.HSP_Q_Seq, 'P',
      db21s.LSBarCode(b.HSP_Midline), '\\|',
      b.HSP_H_Seq, 'P')
    ) AS f
  ) AS y,
  table(SELECT COUNT(*) AS m FROM table(
    db21s.LSMultiMatch3(
      b.HSP_Q_Seq, '.',
      db21s.LSBarCode(b.HSP_Midline), '\\|',
      b.HSP_H_Seq, '.')
    ) AS f
  ) AS z
WHERE float(p) / float(m) < 0.25;
```

In dieser Abfrage ist die BLAST-Ausgabe (BlastOutput) eigentlich eine Sicht eines BLAST-Kurznamens. Die Abfrage verwendet die Funktion 'LSMultiMatch3', um die genauen Übereinstimmungen beim Alignment zurückzugeben. Bei der ersten Verwendung werden die genauen Übereinstimmungen für Symbol „P“ zurückgegeben, bei der zweiten Verwendung werden alle genauen Übereinstimmungen zurückgegeben. Tabelle 115 zeigt eine Zeile aus der Ergebnistabelle.

Tabelle 115. Beispielergebniszeile

HSP_Q_SEQ	HSP_H_SEQ	HSP_INFO	PERCENT_PROLINES
NIWDFMQGN...	NIWDFMQGN...	Identities = 80/80 (100%), Positives = 80/80 (100%), Gaps = 0/80 (0%)	+2.5000000000000000E-002

Grundlage der vorstehenden Abfrage ist folgende Publikation: Extending traditional query-based integration approaches for functional characterization of post-genomic data. (2001) Barbara A Eckman, Anthony S Kosky and Leonardo A Laroco Jr. *Bioinformatics* 17(7), 587-601.

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSMultiMatch3' - Beispiel“ auf Seite 531
- „Benutzerdefinierte Funktion 'LSBarCode'“ auf Seite 528

## Benutzerdefinierte Funktion 'LSMultiMatch'

►—DB2LS.LSMultiMatch—(eingabenukleotidsequenz oder eingabepeptidsequenz, muster)————►

#### eingabenukleotidsequenz oder eingabepeptidsequenz

Eine gültige Zeichenfolgedarstellung, die eine Nukleotid- oder Peptidsequenz beschreibt. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

#### muster

Das Muster, das der von der PERL-Sprache angegebenen Grammatik ent-

spricht. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

Der Schemaname lautet DB2LS.

Mit Hilfe der benutzerdefinierten Funktion 'LSMultiMatch' wird für jede Übereinstimmung, die sich in der Eingabesequenz nicht überschneidet, eine Tabelle zurückgegeben. Jede Tabelle besteht aus einer Anfangsposition und dem übereinstimmenden Sequenzfragment.

Das Ergebnis der Funktion wird in einer Tabelle mit zwei Spalten dargestellt. Die erste Spalte enthält eine ganze Zahl zur Darstellung der Anfangsposition einer Übereinstimmung des Musters in der Sequenz. Die zweite Spalte enthält das übereinstimmende Sequenzfragment.

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSMultiMatch' - Beispiel“ auf Seite 530
- „Benutzerdefinierte Funktion 'LSBarcode'“ auf Seite 528
- „Benutzerdefinierte Funktion 'LSMultiMatch3'“ auf Seite 531

## Benutzerdefinierte Funktion 'LSMultiMatch' - Beispiel

Dieses Beispiel sucht nach der Position und den passenden Fragmenten aller sich nicht überschneidender Übereinstimmungen aus der Eingabe.

```
SELECT position, match FROM table
  (LSMultiMatch('match not and non but no match for no or none',
    'no[tn] ')) as f
```

Die Abfrage gibt eine Tabelle zurück, die auf dieser SELECT-Anweisung basiert und die Ergebnisse der Übereinstimmungen zeigt:

*Tabelle 116. Ergebnis der Funktion 'LSMultiMatch' mit mehreren zurückgegebenen Zeilen*

POSITION	MATCH
7	not
15	non

Die Funktion 'LSMultiMatch' gibt die Position und die passende Zeichenfolge aller Übereinstimmungen zurück. Das folgende Beispiel durchsucht 'Entrez Nucleotide' nach Sequenzeinträgen, die ein bestimmtes Motiv enthalten. Die Abfrage druckt die Sequenzkennungen und die übereinstimmenden Sequenzen aus. Die Unter-muster „{0,9}“ am Anfang und am Ende müssen vor und nach der Sequenz bis neun Zeichen übereinstimmen. Die Abfrage druckt auch diese Zeichen.

```
select SequenceKey, Position, Match from GBSeq,
  table(db2ls.LSMultiMatch(Sequence, '{0,9}(ATG|CGC)ACGGGC.{0,9}') )
  as fmatch
  WHERE entrez.contains(KeywordList,
    'Na/K/2C1 cotransporter AND nkcc1 gene') = 1;
```

Die Abfrage gibt folgendes Ergebnis zurück:

*Tabelle 117. Entrez-Suchdaten*

SEQUENCEKEY	POSITION	MATCH
N02B59AE0.04DD4E84	1	TGCTTGGTGATGACGGGGCTACCCCAAC



Tabelle 117. Entrez-Suchdaten (Forts.)

SEQUENCEKEY	POSITION	MATCH
N02B59AE0.04DD4E84	91	GGCCATGTTTCGCACGGGCTCCAGAAGG
N02B59AE0.04DC5EF4	1	TGCTTGGTGATGACGGGCTACCCCAAC
N02B59AE0.04DC5EF4	91	GGCCATGTTTCGCACGGGCTCCAGAAGG

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSMultiMatch'“ auf Seite 529
- „Benutzerdefinierte Funktion 'LSBarCode'“ auf Seite 528
- „Benutzerdefinierte Funktion 'LSMultiMatch3'“ auf Seite 531

## Benutzerdefinierte Funktion 'LSMultiMatch3'

►—DB2LS.LSMultiMatch3—(eingabezeichenfolge1, muster1, eingabezeichenfolge2, muster2, eingabezeichenfolge3, muster3)—◄

**eingabe-zeichenfolgen**

Eine gültige Zeichenfolgedarstellung, die eine Nukleotid- oder Peptidsequenz oder eine HSP\_Midline-Zeichenfolge aus einem BLAST-Alignment beschreibt. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

**muster**

Das Muster, das der von der PERL-Sprache angegebenen Grammatik entspricht. Die Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

Der Schemaname lautet DB2LS.

Mit Hilfe der benutzerdefinierten Funktion 'LSMultiMatch3' werden drei Muster und drei Zeichenfolgen als Eingabe verwendet und alle Positionen zurückgegeben, an denen alle drei Zeichenfolgen mit den entsprechenden Mustern übereinstimmen. Sie können diese benutzerdefinierte Funktion verwenden, um eine Mustererkennung für ein Alignment durchzuführen.

Das Ergebnis der Funktion ist eine Tabelle mit vier Spalten. Die erste Spalte enthält eine ganze Zahl zur Darstellung der Anfangsposition einer Übereinstimmung des Musters in allen Sequenzen. Die Funktion verankert alle Zeichenfolgen zusammen an der ersten Position. Die zweite, dritte und vierte Spalte enthalten die übereinstimmenden Sequenzfragmente.

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSMultiMatch3' - Beispiel“ auf Seite 531
- „Benutzerdefinierte Funktion 'LSMultiMatch'“ auf Seite 529
- „Benutzerdefinierte Funktion 'LSBarCode'“ auf Seite 528

## Benutzerdefinierte Funktion 'LSMultiMatch3' - Beispiel

Das folgende Beispiel verwendet die Funktion, um den Prozentsatz eines bestimmten Aminosäuresymbols unter den von BLAST zurückgegebenen genauen Übereinstimmungen zu berechnen. Hierbei ist zu beachten, dass in diesem Beispiel auch die benutzerdefinierte Funktion 'LSBarCode' aufgerufen wird. Diese Funktion wird



von der Abfrage benötigt, da BLAST in einem Alignment nicht immer eine Folge von Balken („|“) zurückgibt. Das nachstehende Beispiel zeigt dies:

```
Query:      MDYTTGQILTAGNEHQQRNPASLTKLMTGYVVDRAIDSHRITPDDIVTVGRDAWAKDNPV
Alignment:  MDY +G++L GN ++ +PASLTK+MT YVV +A+ + +I D+VTVG+DAWA NP
Target:     MDYASGKVLAEAGNADEKLDPASLTKIMTSYVVVGQALKADKIKLTDMMVTVGKDAWATGNPA
```

Verwenden Sie die Funktion 'LSBarCode' zum Umwandeln der Sequenz, um zu gewährleisten, dass die Ausgabe mit der korrekten Balkenfolge ausgerichtet (aligniert) wird. Die Funktion ersetzt alle Zeichen (außer Leerzeichen und „+“-Zeichen) durch einen vertikalen Balken.

```
SELECT BlastOutput.* , float( p )/ float( m ) AS percent_prolines
FROM BlastOutput b,
  table(SELECT COUNT(*) AS p FROM table(
    db21s.LSMultiMatch3(
      b.HSP_Q_Seq, 'P',
      db21s.LSBarCode(b.HSP_Midline), '\|',
      b.HSP_H_Seq, 'P')
    ) AS f
  ) AS y,
  table(SELECT COUNT(*) AS m FROM table(
    db21s.LSMultiMatch3(
      b.HSP_Q_Seq, '.',
      db21s.LSBarCode(b.HSP_Midline), '\|',
      b.HSP_H_Seq, '.')
    ) AS f
  ) AS z
WHERE float(p) / float(m) < 0.25;
```

In dieser Abfrage ist die BLAST-Ausgabe (BlastOutput) eine Sicht einer BLAST-Auswahl (SELECT). Die Abfrage verwendet die Funktion 'LSMultiMatch3', um die perfekten Übereinstimmungen beim Alignment zurückzugeben. Bei der ersten Verwendung werden die perfekten Übereinstimmungen für Symbol „P“ zurückgegeben, bei der zweiten Verwendung werden alle perfekten Übereinstimmungen zurückgegeben. Tabelle 118 zeigt eine Zeile aus der Ergebnistabelle.

Tabelle 118. Beispielergebniszeile

HSP_Q_SEQ	HSP_H_SEQ	HSP_INFO	PERCENT_PROLINES
NIWDFMQG...	NIWDFMQG...	Identities = 80/80 (100%), Positives = 80/80 (100%), Gaps = 0/80 (0%)	+2.500000000000000E-002

Grundlage der vorstehenden Abfrage ist folgende Publikation: Extending traditional query-based integration approaches for functional characterization of post-genomic data. (2001) Barbara A Eckman, Anthony S Kosky and Leonardo A Laroco Jr. *Bioinformatics* 17(7), 587-601.

Das folgende Beispiel sucht nach drei getrennten Mustern in drei getrennten Zeichenfolgefsegmenten:

```
SELECT position, match_1, match_2, match_3
FROM table(db21s.LSMultiMatch3('zaza', 'a', 'abab',
  'b', 'bcbc', 'c')) as f
```

Die Abfrage gibt die Positionen und die passenden Zeichenfolgen aller Übereinstimmungen zurück, wie aus der folgenden Tabelle ersichtlich wird:

Tabelle 119. Ergebnis einer Mehrfachübereinstimmung mit drei Eingabemustern

POSITION	MATCH_1	MATCH_2	MATCH_3
2	a	b	c
4	a	b	c

Das nächst Beispiel sucht drei getrennte Muster in drei getrennten Zeichenfolgefragmenten:

```
SELECT position, match_1, match_2, match_3
FROM table
(LSMultiMatch3('cbccbcbccbbcccccbbbbbcccc', 'c{1,3}b{1,3}c{1,3}',
'abcdefghijklmnopqrstuvwxyzabcdefghijklmnopqrstuvwxyz',
'.', '0123456789012345678901234567890123456789', '\d')) as f
```

Die Ergebnisse werden in nachfolgender Tabelle dargestellt:

Tabelle 120. Ergebnis einer Mehrfachübereinstimmung mit drei Eingabemustern

POSITION	MATCH_1	MATCH_2	MATCH_3
1	cbcc	a	0
7	ccbcbcc	g	6

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSBarCode' - Beispiel“ auf Seite 528
- „Benutzerdefinierte Funktion 'LSBarCode'“ auf Seite 528
- „Benutzerdefinierte Funktion 'LSMultiMatch3'“ auf Seite 531

## Benutzerdefinierte Umkehrfunktionen

Benutzerdefinierte Umkehrfunktionen kehren Nukleotid- oder Aminosäuresequenzen um.

### Benutzerdefinierte Funktion 'LSRevComp'

►►—DB2LS.LSRevComp—(eingabe-nukleotidsequenz)——————◄◄

**eingabe-nukleotidsequenz**

Eine gültige Zeichenfolgedarstellung, die eine Nukleotidsequenz beschreibt. Die Sequenz kann IUPAC-Ambiguitätscodes enthalten. Eine Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

Der Schemaname lautet DB2LS.

Das Ergebnis der Funktion ist eine Zeichenfolge vom Datentyp VARCHAR mit einer Länge von maximal 32672 Byte. Die Zeichenfolge ist eine Darstellung des reversen (umgekehrten) Komplements der Nukleotidsequenz.

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion 'LSRevComp' - Beispiel“ auf Seite 534
- „Benutzerdefinierte Funktion 'LSRevNuc'“ auf Seite 535
- „Benutzerdefinierte Funktion 'LSRevPep'“ auf Seite 536

## Benutzerdefinierte Funktion 'LSRevComp' - Beispiel

Sie können die Funktion 'LSRevComp' immer dann in einer SQL-Anweisung verwenden, wenn Sie eine integrierte Funktion verwenden würden, die eine Nukleotidsequenz akzeptiert. Beispiel:

```
SELECT DB2LS.LSRevComp(:NucSeq) FROM SYSDDUMMY1;
```

In diesem Beispiel wird die Funktion verwendet, um das reverse (umgekehrte) Komplement der Eingabesequenz zurückzugeben, die von einer Hostvariablen übergeben wurde.

Bei Verwendung einer ungültigen Zeichenfolge oder eines ungültigen Datentyps wird die folgende Fehlermeldung angezeigt:

```
SQL0443N Die Routine "DB2LS.LSREVCOMP" (spezifischer Name "LSREVCOMP") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Die Sequenz ist ungültig". SQLSTATE=38608
```

Ist das Alphabet der Eingabe nicht korrekt, wird eine Ausnahmebedingung ausgegeben.

Das folgende Beispiel zeigt die Funktionsweise der benutzerdefinierten Funktion 'LSRevComp' in einer Abfrage:

```
SELECT HSP_H_Seq, db21s.LSRevComp(HSP_H_Seq) as REV_HSP_H_Seq
FROM BlastN
WHERE BlastSeq='ccgctagtattggtcaatcttttgatatccaccgaa'
```

Die Abfrageergebnisse werden in nachfolgender Tabelle dargestellt:

HSP_H_SEQ	REV_HSP_H_SEQ
AGTATTGGTCAATCTTTTGAT	ATCAAAAGATTGACCAATACT
TGGTCAATCTTTTGATA	TATCAAAAGATTGACCA
TTGCCAATCTTTTGATATCC	GGATATCAAAAGATTGGCCAA
TCAATCTTTTGATATCC	GGATATCAAAAGATTGA
GGATATCAAAAGATTGA	TCAATCTTTTGATATCC

5 Satz/Sätze ausgewählt.

Sie können die Umkehrfunktion zusammen mit anderen benutzerdefinierten Life Sciences-Funktionen verwenden, um das reverse (umgekehrte) Komplement einer Nukleotidsequenz zu übersetzen, wie aus folgendem Beispiel ersichtlich wird:

```
values db21s.LSNuc2Pep(
      db21s.LSRevComp('TTTTTCTTATTGTCTTCTCATCGTATTTCTTATGTTGCTGATGT'))
```

Die Abfrage gibt folgendes Ergebnis zurück:

```
TSAT*EIR*GRQ*EK
```

### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSRevComp'“ auf Seite 533

## Benutzerdefinierte Funktion 'LSRevNuc'

►—DB2LS.LSRevNuc—(eingabe-nukleotidsequenz)—————►

### eingabe-nukleotidsequenz

Eine gültige Zeichenfolgedarstellung, die eine Nukleotidsequenz beschreibt. Eine Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte. Die Nukleotidsequenz muss ein Teil des DNA-Alphabets sein oder das gesamte DNA-Alphabet darstellen.

Der Schemaname lautet DB2LS.

Das Ergebnis der Funktion ist eine Zeichenfolge vom Datentyp VARCHAR mit einer Länge von maximal 32672 Byte. Die Zeichenfolge ist eine Darstellung der umgekehrten Reihenfolge der Nukleotidsequenz.

### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSRevNuc' - Beispiel“ auf Seite 535
- „Benutzerdefinierte Funktion 'LSRevComp'“ auf Seite 533
- „Benutzerdefinierte Funktion 'LSRevPep'“ auf Seite 536

## Benutzerdefinierte Funktion 'LSRevNuc' - Beispiel

Sie können die Funktion 'LSRevNuc' immer dann in einer SQL-Anweisung verwenden, wenn Sie eine integrierte Funktion verwenden würden, die eine Nukleotidsequenz akzeptiert. Beispiel:

```
SELECT DB2LS.LSRevNuc(:NucSeq) FROM SYSDDUMMY1;
```

In diesem Beispiel wird die Funktion verwendet, um die von einer Hostvariablen übergebenen Eingabedaten umzukehren.

Bei Verwendung einer ungültigen Zeichenfolge oder eines ungültigen Datentyps wird die folgende Fehlermeldung angezeigt:

```
SQL0443N Die Routine "DB2LS.LSREVNUC" (spezifischer Name "LSREVNUC") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Die Sequenz ist ungültig". SQLSTATE=38608
```

Das folgende Beispiel zeigt die Verwendung der benutzerdefinierten Funktion 'LSRevNuc' in einer Abfrage.

```
SELECT HSP_H_Seq, db21s.LSRevNuc(HSP_H_Seq) as REV_HSP_H_Seq
FROM BlastN
WHERE BlastSeq='gtaatacgtagggggctagcgcgggcaactgaagataaagc'
```

Die folgende Ergebnistabelle zeigt die von der Abfrage zurückgegebenen umgekehrten Nukleotidsequenzen:

HSP_H_SEQ	REV_HSP_H_SEQ
CGCGGGCAAAGTGAAGATAAAGC	CGAAATAGAAGTCAAACGGGGCGC
GCGCTAGCCCCCTACGTATTAC	CATTATGCATCCCCCGATCGCG
GTAATACGTAGGGGGCTAGCG	GCGATCGGGGGATGCATAATG
GTAATACGTAGGGGGCTAGCG	GCGATCGGGGGATGCATAATG
GTAATACGTAGGGGGCTAGCG	GCGATCGGGGGATGCATAATG

5 Satz/Sätze ausgewählt.

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSRevNuc'“ auf Seite 535

## Benutzerdefinierte Funktion 'LSRevPep'

►►—DB2LS.LSRevPep—(*eingabe-peptidsequenz*)—◄◄

#### eingabe-peptidsequenz

Eine gültige Zeichenfolgedarstellung, die eine Peptidsequenz beschreibt. Eine Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte. Die Eingabesequenz muss Teil des Proteinalphabets sein.

Der Schemaname lautet DB2LS.

Das Ergebnis der Funktion ist eine Zeichenfolge vom Datentyp VARCHAR mit einer Länge von maximal 32672 Byte. Die Zeichenfolge ist eine Darstellung der umgekehrten Reihenfolge der Peptidsequenz.

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSRevPep' - Beispiel“ auf Seite 536
- „Benutzerdefinierte Funktion 'LSRevComp'“ auf Seite 533
- „Benutzerdefinierte Funktion 'LSRevNuc'“ auf Seite 535

## Benutzerdefinierte Funktion 'LSRevPep' - Beispiel

Sie können die Funktion 'LSRevPep' immer dann in einer SQL-Anweisung verwenden, wenn Sie eine integrierte Funktion verwenden würden, die eine Peptidsequenz akzeptiert. Zum Beispiel:

```
SELECT DB2LS.LSRevPep(:NucSeq) FROM SYSDUMMY1;
```

In diesem Beispiel wird die Funktion verwendet, um die von einer Hostvariablen übergebenen Eingabedaten umzukehren.

Bei Verwendung einer ungültigen Zeichenfolge oder eines ungültigen Datentyps wird die folgende Fehlermeldung angezeigt:

```
SQL0443N Die Routine "DB2LS.LSREVPEP" (spezifischer Name "LSREVPEP") gab einen SQLSTATE-Fehler zurück. Der Diagnosetext lautet "Die Sequenz ist ungültig". SQLSTATE=38608
```

Das folgende Beispiel zeigt die Verwendung der benutzerdefinierten Funktion 'LSRevPep' in einer Abfrage:

```
SELECT HSP_H_Seq, db2ls.LSRevPep(HSP_H_Seq) as REV_HSP_H_Seq
FROM BlastP
WHERE BlastSeq='MLCEIECRALSTAHTRLIHDPEPRDALTYLEGKNIFTEDH'
```

Die folgende Ergebnistabelle zeigt die von der Abfrage zurückgegebenen umgekehrten Peptidsequenzen.

HSP_H_SEQ	REV_HSP_H_SEQ
MLCEIECRALSTAHTRLIHDPEPRDALTYL...	HDETFINKGELYTLADRPEFDHILRTHATS...
RVVSTEHTRLVTDAYPEFSISFTATKN	NKTATFSISFEPYADTVLRTHETSVVR
STAHIRVLRDMVPGDEITCFYGSEFF	FFESGYFCTIEDGPVMDRLVRIHATS
AHTRRCPDHEPRGVITYL	LYTIVGRPEHDPCRRTHA

4 Satz/Sätze ausgewählt.

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSRevPep'“ auf Seite 536

---

## Benutzerdefinierte Übersetzungsfunktionen

Die benutzerdefinierten Übersetzungsfunktionen wandeln eine Nukleotidsequenz in eine Peptidsequenz um.

### Benutzerdefinierte Funktion 'LSNuc2Pep'

→ DB2LS.LSNuc2Pep(*eingabe-nukleotidsequenz*, *dateipfad zur externen übersetzungstabelle*) →

#### eingabe-nukleotidsequenz

Eine gültige Zeichenfolgedarstellung, die eine Nukleotidsequenz beschreibt. Eine Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

#### dateipfad zur externen Übersetzungstabelle

Wenn Sie eine angepasste Übersetzungstabelle verwenden, müssen Sie den entsprechenden Dateipfad angeben, damit die Tabelle gefunden wird. Der Zeichenfolgewert des Pfads darf 255 Zeichen nicht überschreiten.

Der Schemaname lautet DB2LS.

Das Ergebnis der Funktion ist eine Zeichenfolge vom Datentyp VARCHAR mit einer Länge von maximal 10890 Byte. Die Zeichenfolge ist eine Darstellung der Peptidsequenz.

Die Eingabe ist eine Nukleotidsequenz, für die der IUB-Zeichensatz verwendet wird. Bei diesen Funktionen wird davon ausgegangen, dass das erste Codon am ersten Zeichen der Nukleotidsequenz beginnt. Beginnt das erste Codon nicht am ersten Zeichen der Nukleotidsequenz, verwenden Sie für die Eingabesequenz eine SUBSTR-Funktion.

Das Ergebnis der Funktion ist eine Peptidsequenz mit den standardmäßigen Aminosäuresymbolen.

Die Funktion

- entfernt Leerzeichen in Eingabesequenzen;
- ignoriert nicht zugehörige Nukleotide außerhalb eines Leserahmens;
- gibt eine Leerausgabe zurück, wenn eine leere Nukleotidsequenz als Eingabe verwendet wird.

### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSNuc2Pep' - Beispiel“ auf Seite 538
- „Benutzerdefinierte Funktion 'LSTransAllFrames“ auf Seite 539

## Benutzerdefinierte Funktion 'LSNuc2Pep' - Beispiel

Angenommen, Sie wollen die Daten von Nukleotidsequenzen in eine Peptidsequenz übersetzen. Bei diesem Beispiel wird davon ausgegangen, dass das erste Codon am ersten Zeichen der Nukleotidsequenz beginnt.

Sie können die Funktion mit einer values-Anweisung aufrufen. Die einzige Eingabe ist eine Nukleotidsequenz, wie in folgendem Beispiel gezeigt:

```
values db2ls.LSNuc2Pep('TTTTTCTTATTGTCTTCCTCATCGTATTTCTTATGTTGCTGATGT')
```

Das Ergebnis der vorstehenden Anweisung ist eine Peptidsequenz mit den standardmäßigen Aminosäuresymbolen:

```
FLLSSSSYFLCC*C
```

Wenn Sie die Übersetzung im Leserahmen +2 wünschen, verwenden Sie folgendes Beispiel:

```
values LSNuc2Pep(SUBSTR('TTTTTCTTATTGTCTTCCTCATCGTATTTCTTATGTTGCTGATGT',2))
```

Die ganze Zahl in der Anweisung gibt die Anfangsposition der Suche nach dem Codon an.

Es folgt ein Beispiel zur Verwendung dieser Funktion als Vergleichselement in einer Abfrage.

```
SELECT *
  FROM proteindata
 WHERE peptideseq=DB2LS.LSNuc2Pep('TTTTTCTTATTGTCTTCCTCATCG
                                     TATTTCTTATGTTGCTGATGT');
```

Das Ergebnis wird in Tabelle 121 gezeigt.

*Tabelle 121. Ergebnisse mit Funktion 'LSNuc2Pep' als Vergleichselement*

ID	PROTEINNAME	PEPTIDSEQ
1	proteinA	FSYCLPHRISYVAD

Das folgende Beispiel übersetzt eine Nukleotidsequenz anhand einer externen Übersetzungstabelle in eine Peptidsequenz. Der erste Parameter ist die Nukleotidsequenz, und der zweite Parameter ist der Pfad zur externen Übersetzungstabelle.

```
values db2ls.LSNuc2Pep('TTTTCTTATTGTCTTCCTCATCGTATTTCTTATGTTGCTGATGT',
                        'C:\translation.txt')
```

Das Ergebnis der vorstehenden Anweisung mit der angegebenen Übersetzungstabelle ist die folgende Zeichenfolge:

```
FSYCLPHRISYVAD
```

In folgendem Beispiel werden zwei der benutzerdefinierten Funktionen kombiniert, um den zusätzlichen Nutzen der Funktionen zu demonstrieren:

```
values DB2LS.LSNuc2Pep(DB2LS.LSRevCompNuc('TTT..'))
```

Hierbei ist zu beachten, dass das vorstehende Beispiel dieselben Ergebnisse zurückgibt wie die folgende Abfrage:

```
select * from table (DB2LS.LSTransAllFrames ('TTT..')) as t
where t.readframe = -1
```

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSRevNuc' - Beispiel“ auf Seite 535
- „Benutzerdefinierte Funktion 'LSTransAllFrames' - Beispiel“ auf Seite 540
- „Benutzerdefinierte Funktion 'LSNuc2Pep'“ auf Seite 537

## Benutzerdefinierte Funktion 'LSTransAllFrames'

► DB2LS.LSTransAllFrames—(eingabe-nukleotidsequenz—,dateipfad zur externen übersetzungstabelle—)►

#### eingabe-nukleotidsequenz

Eine gültige Zeichenfolgedarstellung, die eine Nukleotidsequenz beschreibt. Die Eingabesequenz kann IUPAC-Ambiguitätscodes enthalten. Eine Zeichenfolgedarstellung muss über den Datentyp VARCHAR verfügen und darf nicht länger sein als 32672 Byte.

#### dateipfad zur externen Übersetzungstabelle

Wenn Sie eine angepasste Übersetzungstabelle verwenden, müssen Sie den entsprechenden Dateipfad angeben, damit die Tabelle gefunden wird. Der Zeichenfolgewert des Pfads darf 255 Zeichen nicht überschreiten.

Der Schemaname lautet DB2LS.

Verwenden Sie die benutzerdefinierte Funktion 'LSTransAllFrames', um aus einer angegebenen Nukleotidsequenz eine Gruppe von Peptidsequenzen zu generieren. Diese Peptidsequenzen stellen mögliche Übersetzungen der Eingabe-Nukleotidsequenz in jedem der 6 Rahmen dar. Diese Funktion ist nützlich, wenn die Eingabe Fehler enthält oder der Leserahmen unbekannt ist.

Das Ergebnis der Funktion wird in einer Tabelle mit zwei Spalten dargestellt. Die erste Spalte ist mit READFRAME markiert und stellt den Rahmen für die Übersetzung dar. Diese Spalte enthält eine ganze Zahl, die die Anfangsposition der Übersetzung darstellt. Eine negative ganze Zahl gibt eine Übersetzung des entgegengesetzten Strangs an. Die zweite Spalte namens PEPTID ist eine Zeichenfolge vom Datentyp VARCHAR mit einer Länge von maximal 10890 Byte. Die Zeichenfolge ist eine Darstellung der Peptidsequenz.

Die Funktion

- entfernt Leerzeichen in Eingabesequenzen;
- ignoriert nicht zugehörige Nukleotide außerhalb eines Leserahmens;
- gibt eine Leerausgabe zurück, wenn eine leere Nukleotidsequenz als Eingabe verwendet wird.

#### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSTransAllFrames' - Beispiel“ auf Seite 540
- „Benutzerdefinierte Funktion 'LSNuc2Pep'“ auf Seite 537



## Benutzerdefinierte Funktion 'LSTransAllFrames' - Beispiel

Angenommen, Sie wollen eine Nukleotidsequenz mit Hilfe der integrierten Übersetzungstabelle in allen sechs Leserahmen (Readframes) übersetzen. Das folgende Beispiel zeigt, wie dies funktioniert:

```
SELECT * FROM table(DB2LS.LSTransAllFrames('TTTTTCTTATTGTCTTCCTCATCG
                                           TATTTCTTATGTTGCTGATGT')) as t;
```

Die Abfrage gibt die Peptide in einer Tabelle zurück, wie aus folgendem Beispiel ersichtlich wird:

*Tabelle 122. Ergebnis der Übersetzung einer Nukleotidsequenz*

READFRAME	PEPTIDE
1	FLLSSSSYFLCC*C
2	FSYCLPHRISYVAD
3	FLIVFLIVFLMLLM
-1	TSAT*EIR*GRQ*EK
-2	HQQHKKYDEEDNKK
-3	ISNIRNTMRKTIRK

Das nächste Beispiel verwendet eine angepasste Übersetzungstabelle, um eine Nukleotidsequenz in allen sechs Leserahmen zu übersetzen.

```
SELECT * FROM table
  (DB2LS.LSTransAllFrames
   ('TTTTTCTTATTGTCTTCCTCATCGTATTTCTTATGTTGCTGATGT',
    'C:\msvs6\MyProjects\alin_udf\test\files\translation.txt')) as t;
```

Die Ergebnistabelle ist die gleiche wie im vorherigen Beispiel, da die Eingabesequenz sich nicht geändert hat und die Übersetzungstabelle der in die Funktion integrierten Tabelle entspricht.

In folgendem Beispiel werden zwei der benutzerdefinierten Funktionen kombiniert, um den zusätzlichen Nutzen der Funktionen zu demonstrieren:

```
values DB2LS.LSNuc2Pep(DB2LS.LSRevCompNuc('TTT..'))
```

Hierbei ist zu beachten, dass das vorstehende Beispiel dieselben Ergebnisse zurückgibt wie die folgende Abfrage:

```
select * from table (DB2LS.LSTransAllFrames ('TTT..')) as t
where t.readframe = -1
```

Das folgende Beispiel wählt aus der von der Funktion 'LSTransAllFrames' generierten Ausgabe einen bestimmten Leserahmen aus.

```
SELECT * FROM
  TABLE(db2ls.LSTransAllFrames('TTTTTCTTATTGTCTTCCTCATCG
                                TATTTCTTATGTTGCTGATGT')) AS t
WHERE t.readframe=-2
```

Die Abfrage gibt folgendes Ergebnis zurück:

*Tabelle 123. Verwendung der Leserahmenfunktion*

READFRAME	PEPTIDE
-2	HQQHKKYDEEDNKK

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion ‘LSNuc2Pep’ - Beispiel“ auf Seite 538
- „Benutzerdefinierte Funktion ‘LSRevNuc’ - Beispiel“ auf Seite 535
- „Benutzerdefinierte Funktion ‘LSTransAllFrames’“ auf Seite 539

---

## Format der Codon-Frequenztafel

Eine Codon-Frequenztafel zeigt die Frequenz, mit der die Aminosäuren in ein bestimmtes Codon zurückübersetzt werden. Die benutzerdefinierte Funktion ‘LSPep2ProbNuc’ verwendet die Codon-Frequenztafel, um aus einer angegebenen Peptidsequenz eine Nukleotidsequenz zu ermitteln.

Die folgende Liste erläutert das Format der Datei mit der Codon-Frequenztafel:

- Zwei benachbarte Punkte markieren den Anfang der Tafel. Alle Textdaten, die vor diesen beiden Punkten stehen, sind Kommentare. Die beiden benachbarten Punkte sind auch dann erforderlich, wenn davor keine Kommentare stehen.
- Die Tafel enthält die folgenden Spalten:
  1. Am-Acid: Ein aus drei Buchstaben bestehender Code für das Aminosäuresymbol.
  2. Codon: Das Codon für das betreffende Aminosäuresymbol.
  3. Number: Die Häufigkeit, mit der das betreffende Codon in den Genen vorkommt, aus denen die Tafel aufgebaut ist.
  4. x/1000: Die erwartete Häufigkeit des Aminosäure-Codon-Paars pro 1000 Übersetzungen in Genen.
  5. Fraction: Der Bruchteil der Häufigkeit des Codons in seiner synonymen Codon-Familie.

Beispiele für Codon-Frequenztafeln stehen im Unterverzeichnis ‘`sqlib/samples/lifesci/ls_udfs`’ zur Verfügung.

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion ‘LSPep2ProbNuc’“ auf Seite 511
- „Codon-Frequenztafel - Beispiel“ auf Seite 542

---

## Codon-Frequenztafel - Beispiel

Abb. 46 zeigt anhand eines Beispiels das Format einer Codon-Frequenztafel.

---

Am-Acid	Codon	Number	x/1000	Fraction	..
Gly	GGG	198.00	18.34	0.23	
Gly	GGA	71.00	6.58	0.08	
Gly	GGT	66.00	6.11	0.08	
Gly	GGC	527.00	48.81	0.61	
Glu	GAG	534.00	49.46	0.88	
Glu	GAA	71.00	6.58	0.12	
Asp	GAT	31.00	2.87	0.06	
Asp	GAC	481.00	44.55	0.94	
Val	GTG	396.00	36.68	0.47	
Val	GTA	22.00	2.04	0.03	
Val	GTT	44.00	4.08	0.05	
Val	GTC	384.00	35.57	0.45	
Ala	GCG	446.00	41.31	0.39	
Ala	GCA	71.00	6.58	0.06	
Ala	GCT	116.00	10.74	0.10	
Ala	GCC	503.00	46.59	0.44	
... (truncated)					

---

Abbildung 46. Beispiel einer Codon-Frequenztafel

### Zugehörige Referenzen:

- „Benutzerdefinierte Funktion 'LSPep2ProbNuc'“ auf Seite 511
- „Format der Codon-Frequenztafel“ auf Seite 541

---

## Format von Übersetzungstabellen

Dieser Abschnitt erläutert das Format von Übersetzungstabellen, die von den benutzerdefinierten Life Sciences-Funktionen 'LSPep2AmbNuc', 'LSTransAllFrames' und 'LSNuc2Pep' verwendet werden.

Die folgende Liste erläutert das Format der Datei mit der Codon-Frequenztafel:

- Zwei benachbarte Punkte markieren den Anfang der Tafel. Alle Textdaten, die vor diesen beiden Punkten stehen, sind Kommentare.
- Jede Zeile der Tafel besteht aus einem einbuchstabigen Aminosäuresymbol, dem dreibuchstabigen Namen der Aminosäure, den eindeutigen Codons, einem Ausrufezeichen und den mehrdeutigen Codons. Die einzelnen Wörter in der Zeile werden jeweils durch ein Leerzeichen getrennt.
- Jedes Codon und jedes Aminosäuresymbol darf in der Datei jeweils nur einmal vorkommen.
- Stoppcodons werden in das Symbol '\*' übersetzt.
- Codons, die aus Kleinbuchstaben bestehen, sind Startcodons.

- Alle anderen Codons bestehen aus Großbuchstaben.
- Codons, für die es keine Übersetzung in ein entsprechendes Aminosäuresymbol gibt, werden in das Symbol 'X' übersetzt.

Beispiele für Übersetzungstabellen stehen im Unterverzeichnis 'sqlib/samples/lifesci/ls\_udfs' zur Verfügung.

**Zugehörige Konzepte:**

- „Benutzerdefinierte Life Sciences-Funktionen - Übersicht“ auf Seite 503

**Zugehörige Referenzen:**

- „Übersetzungstabelle - Beispiel“ auf Seite 543

## Übersetzungstabelle - Beispiel

Abb. 47 zeigt das Format einer Beispielübersetzungstabelle.

Beispielübersetzungstabelle					
Symbol	3 Buchstaben	Codons	!	IUPAC	..
A	Ala	GCT GCC GCA GCG	!	GCX	
B	Asx		!	RAY	
C	Cys	TGT TGC	!	TGY	
D	Asp	GAT GAC	!	GAY	
E	Glu	GAA GAG	!	GAR	
F	Phe	TTT TTC	!	TTY	
G	Gly	GGT GGC GGA GGG	!	GGX	
H	His	CAT CAC	!	CAY	
I	Ile	ATT ATC ATA	!	ATH	
K	Lys	AAA AAG	!	AAR	
L	Leu	TTG TTA CTT CTC CTA CTG	!	TTR CTX YTR	; YTX
M	Met	atg	!	ATG	
N	Asn	AAT AAC	!	AAY	
P	Pro	CCT CCC CCA CCG	!	CCX	
Q	Gln	CAA CAG	!	CAR	
R	Arg	CGT CGC CGA CGG AGA AGG	!	CGX AGR MGR	; MGX
S	Ser	TCT TCC TCA TCG AGT AGC	!	TCX AGY	; WSX
T	Thr	ACT ACC ACA ACG	!	ACX	
V	Val	GTT GTC GTA GTG	!	GTX	
W	Trp	TGG	!	TGG	
X	Xxx		!	XXX	
Y	Tyr	TAT TAC	!	TAY	
Z	Glx		!	SAR	
*	End	TAA TAG TGA	!	TAR TRA	; TRR

Abbildung 47. Beispielübersetzungstabelle

**Zugehörige Konzepte:**

- „Benutzerdefinierte Life Sciences-Funktionen - Übersicht“ auf Seite 503

**Zugehörige Referenzen:**

- „Format von Übersetzungstabellen“ auf Seite 542



---

## Kapitel 26. Benutzerdefinierte KEGG-Funktionen

In diesem Kapitel wird erläutert, was die benutzerdefinierten KEGG-Funktionen sind, wie sie einem System zusammengeschlossener Datenbanken hinzugefügt und in Abfragen verwendet werden.

---

### Benutzerdefinierte KEGG-Funktionen - Übersicht

Die "Kyoto Encyclopedia of Genes and Genomes" (KEGG) stellt eine Gruppe von Datenbanken dar, die Genominformationen enthalten. Die benutzerdefinierten KEGG-Funktionen sind ein Satz Funktionen, der mit DB2<sup>®</sup> Information Integrator für den Zugriff auf Genominformationen in KEGG-Datenbanken bereitgestellt wird.

Die Datenbank für Reaktionspfade (Pathway-Datenbank) und die Sequence Similarity-Datenbank (Datenbank für die Sequenzsimilaritätssuche, SSDB) sind die einzigen beiden Datenbanken in der KEGG-Suite, auf die DB2 Information Integrator über die Schnittstelle der KEGG-Web-Services zugreifen kann. Die Datenbank für Reaktionspfade ist eine Datensammlung über Netzwerke molekularer Interaktionen in biologischen Prozessen, einschließlich metabolischer, regulierender und molekularer Reaktionspfade. Die Sequence Similarity-Datenbank (SSDB) ist eine Datensammlung über Protein codierende Gene in den gesamten Genomkomplexen.

Die benutzerdefinierten KEGG-Funktionen verwenden die KEGG-API, um auf diese Datenbanken zuzugreifen.

Einige der KEGG-Methoden geben Listen mit Werten zurück, z. B. Genen oder Reaktionspfaden. Für einige dieser Methoden sind Listen mit Werten auch als Eingabe erforderlich. Um die Zusammenstellung komplexer Operationen mit mehreren Methoden zu erleichtern, sind die meisten der benutzerdefinierten KEGG-Funktionen sowohl im Tabellen- als auch im Skalarformat vorhanden. Die Tabellenfunktionen geben eine Tabelle mit einzelnen Werten zurück. Die Skalarfunktionen geben Werte in Form einer durch Leerzeichen unterteilten Liste zurück.

Die benutzerdefinierten KEGG-Funktionen werden mit der Komponente der benutzerdefinierten Life Sciences-Funktionen der nicht relationalen Wrapper installiert. Nachdem die benutzerdefinierten KEGG-Funktionen installiert sind, müssen Sie die Funktionen registrieren.

Um Namensbereichskonflikte zu vermeiden, werden alle benutzerdefinierten KEGG-Funktionen im Schema DB2LS registriert.

#### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548
- „Inaktivieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 577

#### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie

DB2 Information Integrator beinhaltet benutzerdefinierte KEGG-Funktionen für den Zugriff auf Daten in der Datenbank für Reaktionspfade (Pathway-Datenbank) und in der Sequence Similarity-Datenbank (Datenbank für die Sequenzsimilaritätssuche, SSDB).

Die folgende Tabelle listet benutzerdefinierte Funktionen zum Abrufen von Daten aus der Datenbank für Reaktionspfade auf.

*Tabelle 124. Benutzerdefinierte Funktionen für Reaktionspfade*

Beschreibung	Funktionsstypen	Funktionsnamen
Verbindungen nach Reaktionspfad	Skalarfunktion	CompoundsByPathwyS
	Tabellenfunktion	CompoundsByPathwyT
Enzyme nach Reaktionspfaden	Skalarfunktion	EnzymesByPathwyS
	Tabellenfunktion	EnzymesByPathwyT
Gene nach Reaktionspfaden	Skalarfunktion	GenesByPathwyS
	Tabellenfunktion	GenesByPathwyT
Reaktionspfade nach Verbindung	Skalarfunktion	PathwysByCompndsS
	Tabellenfunktion	PathwysByCompndsT
Reaktionspfade nach Enzymen	Skalarfunktion	PathwysByEnzymesS
	Tabellenfunktion	PathwysByEnzymesT
Reaktionspfade nach Genen	Skalarfunktion	PathwysByGenesS
	Tabellenfunktion	PathwysByGenesT

Die folgende Tabelle listet benutzerdefinierte Funktionen zum Abrufen von Daten aus der SSDB-Datenbank auf.

*Tabelle 125. Benutzerdefinierte SSDB-Funktionen*

Beschreibung	Funktionsstypen	Funktionsnamen
Alle Nachbarn nach Genen	Skalarfunktion	AllNbrsByGeneS
	Tabellenfunktion	AllNbrsByGeneT
Beste Nachbarn (Best Neighbors) nach Genen	Skalarfunktion	BestNbrsByGeneS
	Tabellenfunktion	BestNbrsByGeneT
Starke beste Nachbarn (Best-Best Neighbors) nach Genen	Skalarfunktion	BstBstNbrsByGeneS
	Tabellenfunktion	BstBstNbrsByGeneT
Inverse beste Nachbarn (Reverse Best Neighbors) nach Genen	Skalarfunktion	RevBestNbrsByGeneS
	Tabellenfunktion	RevBestNbrsByGeneT
Homologe nach Genen	Skalarfunktion	BestHmlgsByGenesS
	Tabellenfunktion	BestHmlgsByGenesT
Starke beste homologe nach Genen	Skalarfunktion	BstBstHmlgByGenesS
	Tabellenfunktion	BstBstHmlgByGenesT
Paraloge nach Genen	Skalarfunktion	ParalogsByGeneS
	Tabellenfunktion	ParalogsByGeneT
Definitionen nach Genen	Skalarfunktion	DefinitionsByGeneS
Gene nach Motiven	Tabellenfunktion	GenesByMotifsT
Smith-Waterman-Score zwischen Genen	Skalarfunktion	ScoreBetweenGenesS

Alle benutzerdefinierten Tabellenfunktionen für die SSDB-Datenbank geben eine fixierte Gruppe von Ausgabespalten zurück (mit Ausnahme der Funktion `GetGenesByMotifsT`). Die Funktion `GetGenesByMotifsT` gibt *keggid* VARCHAR(100) und *definition* VARCHAR(1000) für jedes Gen zurück.

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545
- „Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten“ auf Seite 560

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

---

## Funktionsargumente für benutzerdefinierte KEGG-Funktionen

Die benutzerdefinierten KEGG-Funktionen verwenden einen Satz allgemeiner Argumente. Die Argumente für die benutzerdefinierten KEGG-Funktionen werden in der folgenden Liste beschrieben.

**cpdlist**

Eine Liste der Verbindungen. Diese Liste muss über den Datentyp VARCHAR und eine tatsächliche Länge von nicht mehr als 1630 Byte verfügen. Das Format für jeden Eintrag in der Liste ist wie folgt:

*cpd:verbindung*

*verbindung* ist die Kennung der Verbindung. Jeder Eintrag der Liste muss durch Leerzeichen, Kommata oder Semikolons begrenzt sein.

**enzymelist**

Eine Liste der Enzyme. Diese Liste muss über den Datentyp VARCHAR und eine tatsächliche Länge von nicht mehr als 1630 Byte verfügen. Jeder Eintrag der Liste muss durch Leerzeichen, Kommata oder Semikolons begrenzt sein.

**keggid**

Eine eindeutige KEGG-ID für jeden Organismus, ausgedrückt als Zeichenfolge. In der KEGG-API wird diese Kennung als 'gene\_ID' bezeichnet.

Die ID muss über den Datentyp VARCHAR und eine tatsächliche Länge von nicht mehr als 100 Byte verfügen. Das Format für das Argument *keggid* lautet wie folgt: *keggorg:gene\_name*. *keggorg* ist ein Code mit drei Buchstaben für einen KEGG-Organismus und wird als Zeichenfolge ausgedrückt. *gene\_name* ist der Name des Gens.

Jedem Organismus in den KEGG-Datenbanken wird eine ID zugeordnet. Die Liste der Organismen ändert sich häufig. Durchsuchen Sie die aktuelle Liste der Genome unter <http://www.genome.ad.jp/kegg/kegg2.html#genes> nach der korrekten ID.

**keggidlist**

Eine Liste der KEGG-IDs. Diese Liste besteht aus einer Zeichenfolge mit begrenzten Werten. Diese Liste muss über den Datentyp VARCHAR und eine tatsächliche Länge von nicht mehr als 1630 Byte verfügen. Jeder Eintrag der Liste muss durch Leerzeichen, Kommata oder Semikolons begrenzt sein.

**orglist** Eine Liste der Namen von KEGG-Organismen. Diese Liste besteht aus einer Zeichenfolge mit begrenzten Werten. Diese Liste muss über den



Datentyp VARCHAR und eine tatsächliche Länge von nicht mehr als 1630 Byte verfügen. Jeder Eintrag der Liste muss durch Leerzeichen, Kommata oder Semikolons begrenzt sein.

Wird dieser Parameter nicht angegeben, wird die gesamte Liste der Organismen durchsucht.

#### **midlist**

Eine Liste der Motivkennungen. *midlist* ist eine Zeichenfolge, die aus begrenzten Werten besteht. Diese Liste muss über den Datentyp VARCHAR und eine tatsächliche Länge von nicht mehr als 1630 Byte verfügen. Jede Motivkennung in der Liste 'midlist' muss durch Leerzeichen, Kommata oder Semikolons begrenzt sein.

Das Format für eine Motivkennung lautet: *database:entry*. Die Datenbank ist die Kennung für eine Motivdatenbank, und der Eintrag ist ein Motiveintrag in dieser Datenbank. Die Motivkennung für den Eintrag 'DnaJ' in der Datenbank 'pfam' lautet z. B. *pf:DnaJ*.

#### **pathwayid**

Ein dreiteiliger Name, der einen bestimmten Reaktionspfad angibt. Dieser Name ist eine Zeichenfolge. Er muss über den Datentyp VARCHAR und eine tatsächliche Länge von nicht mehr als 100 Byte verfügen. Das Format für *pathwayid* lautet:

*path:org zuordnung*

*org* ist die Kennung eines KEGG-Organismus, und *map* ist die Kennung einer Reaktionspfadzuordnung.

#### **threshold**

Der Wert für ein Smith-Waterman-Score. Der Wert muss größer gleich 100 sein und über den Datentyp INTEGER verfügen.

#### **Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

#### **Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

#### **Zugehörige Referenzen:**

- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

---

## **Registrieren der benutzerdefinierten KEGG-Funktionen**

Vor Verwendung der benutzerdefinierten KEGG-Funktionen müssen diese registriert werden.

Der Befehl 'enable\_KEGGFunctions' registriert alle benutzerdefinierten KEGG-Funktionen in der zusammengeschlossenen Datenbank. Die Funktionen werden mit dem Schemanamen DB2LS registriert.

#### **Voraussetzungen:**

Die Komponente der benutzerdefinierten Life Sciences-Funktionen der nicht relationalen Wrapper muss mit DB2 Information Integrator installiert werden.

## Vorgehensweise:

Um die benutzerdefinierten KEGG-Funktionen zu registrieren, führen Sie den Befehl 'enable\_KEGGFunctions' aus.

- Auf Servern mit zusammengeschlossenen Datenbanken, die unter Windows ausgeführt werden, befindet sich dieser Befehl im Verzeichnis `sql\lib\bin`.
- Auf Servern mit zusammengeschlossenen Datenbanken, die unter UNIX ausgeführt werden, befindet sich dieser Befehl im Verzeichnis `sql\lib\bin`.

Syntax:

```
enable_KEGGFunctions -n dbName -u benutzerID -p kennwort [-force]
```

### **dbName**

Der Name der zusammengeschlossenen Datenbank, in der Sie die Funktionen registrieren.

### **benutzer-id**

Eine gültige Benutzer-ID für die zusammengeschlossene Datenbank.

### **kennwort**

Ein gültiges Kennwort für die Benutzer-ID.

**force** Eine optionale Markierung, mit der die Funktionen entfernt und erneut registriert werden können. Verwenden Sie diese Markierung, falls die Funktionen beschädigt oder versehentlich gelöscht werden.

## Beispiel für das Registrieren der benutzerdefinierten Funktionen:

Das folgende Beispiel zeigt die Ausgabe, die beim Absetzen des Befehls 'enable\_KEGGFunctions' zurückgegeben wird:

```
C:> enable_KEGGFunctions -n federateddb -u db2admin -p db2admin
```

```
(0) KEGG Functions were found
-- Create KEGG Functions ...
Create KEGG Functions Successfully.
```

```
*** Please allow a few seconds to clean up the system .....
```

## Beispiel für die Verwendung der Markierung 'force' zum Löschen und Registrieren der benutzerdefinierten Funktionen:

Das folgende Beispiel zeigt die Ausgabe, die beim Absetzen des Befehls 'enable\_KEGGFunctions' mit der Markierung 'force' zurückgegeben wird, wenn die benutzerdefinierten Funktionen bereits registriert sind:

```
C:> enable_KEGGFunctions -n federateddb -u db2admin -p db2admin -force
```

```
(37) KEGG Functions were found

KEGG functions already exist ...
Reinstall KEGG functions ...
-- Drop KEGG Functions ...
Drop KEGG Functions Successfully.
-- Create KEGG Functions ...
Create KEGG Functions Successfully.
```

```
*** Please allow a few seconds to clean up the system .....
```

#### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

#### Zugehörige Tasks:

- „Hinzufügen von relationalen Wrappern, nicht relationalen Wrappern und benutzerdefinierten Funktionen zu DB2 Information Integrator-System“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Inaktivieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 577

#### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

---

## Pathway-Datenbankfunktionen

In den nachstehenden Abschnitten werden die benutzerdefinierten Funktionen für die Pathway-Datenbank erläutert. Diese Abschnitte enthalten Informationen zur Syntax sowie Beispiele der einzelnen Funktionen.

### Benutzerdefinierte Funktion GenesByPathwyS

Verwenden Sie die Funktion GenesByPathwyS, um nach allen Genen in einem Reaktionspfad zu suchen.

Die Funktion GenesByPathwyS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Genen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

#### Syntax:

►►—DB2LS.GenesByPathwyS—(*pathwayid*)—◄◄

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

#### Beispiel für das Suchen aller Gene in einem Reaktionspfad:

Um nach allen Genen in einem bestimmten Reaktionspfad zu suchen, geben Sie das Argument *pathwayid* in der Funktion an.

Beispielsweise möchten Sie nach allen E.coli-Genen in einem Reaktionspfad 00020 suchen.

Dann verwenden Sie folgende Klausel:

```
VALUES CAST(DB2LS.GenesByPathwyS
           ('path:eco00020')
           AS VARCHAR(1000));
```

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion GenesByPathwyT

Verwenden Sie die Funktion GenesByPathwyT, um nach allen Genen in einem Reaktionspfad zu suchen.

Die Funktion GenesByPathwyT ist eine Tabellenfunktion, die eine Spalte VARCHAR(100) mit den Namen der Gene im Reaktionspfad zurückgibt. Der Name der zurückgegebenen Spalte ist gene.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

**Syntax:**

►►—DB2LS.GenesByPathwyT—(*pathwyid*)—◀◀

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

**Beispiel für das Suchen aller Gene in einem Reaktionspfad:**

Um nach allen Genen in einem bestimmten Reaktionspfad zu suchen, geben Sie das Argument *pathwyid* in der Funktion an.

Beispielsweise möchten Sie nach allen E.coli-Genen in einem Reaktionspfad 00020 suchen.

Dann lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.GenesByPathwyT  
    ('path:eco00020')) AS t;
```

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion `CompoundsByPathwyS`

Verwenden Sie die Funktion `CompoundsByPathwyS`, um nach allen Verbindungen in einem Reaktionspfad zu suchen.

Die Funktion `CompoundsByPathwyS` ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Verbindungen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp `VARCHAR`, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung `SELECT` verwendet werden.

### Syntax:

```
►► DB2LS.CompoundsByPathwyS(pathwyid) ◀◀
```

Der Schemaname für diese benutzerdefinierte Funktion ist `DB2LS`.

### Beispiel für das Suchen aller Komponenten in einem Reaktionspfad:

Um nach allen Verbindungen in einem Reaktionspfad zu suchen, geben Sie das Argument *pathwyid* in der Funktion an.

Beispielsweise möchten Sie nach allen Verbindungen in einem Reaktionspfad 00020 suchen.

Dann verwenden Sie folgende Klausel:

```
VALUES CAST(DB2LS.CompoundsByPathwyS  
           ('path:00020')  
           AS VARCHAR(1000));
```

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion `CompoundsByPathwyT`

Verwenden Sie die Funktion `CompoundsByPathwyT`, um nach allen Verbindungen in einem Reaktionspfad zu suchen.

Die Funktion `CompoundsByPathwyT` ist eine Tabellenfunktion, die eine Spalte `VARCHAR(100)` mit den Namen der Verbindungen im Reaktionspfad zurückgibt. Der Name der zurückgegebenen Spalte ist `compound`.

Diese Funktion kann in einer Anweisung `SELECT` verwendet werden.

### Syntax:

►►—DB2LS.CompoundsByPathwyT—(*pathwayid*)—◀◀

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen aller Komponenten in einem Reaktionspfad:

Um nach allen Verbindungen in einem Reaktionspfad zu suchen, geben Sie das Argument *pathwayid* in der Funktion an.

Beispielsweise möchten Sie nach allen Verbindungen in einem Reaktionspfad 00020 suchen.

Dann lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.CompoundsByPathwyT
  ('path:00020')) AS t;
```

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion EnzymesByPathwyS

Verwenden Sie die Funktion EnzymesByPathwyS, um nach allen Enzymen in einem Reaktionspfad zu suchen.

Die Funktion EnzymesByPathwyS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Enzymen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

►►—DB2LS.EnzymesByPathwyS—(*pathwayid*)—◀◀

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen nach allen Enzymen in einem Reaktionspfad:

Um nach allen Enzymen in einem Reaktionspfad zu suchen, geben Sie das Argument *pathwayid* in der Funktion an.

Beispielsweise möchten Sie nach allen Enzymen in einem Reaktionspfad 00020 suchen.

Dann verwenden Sie folgende Klausel:

```
VALUES CAST(DB2LS.EnzymesByPathwyS  
          ('path:00020')  
          AS VARCHAR(1000));
```

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion EnzymesByPathwyT

Verwenden Sie die Funktion EnzymesByPathwyT, um nach allen Enzymen in einem Reaktionspfad zu suchen. Die Funktion EnzymesByPathwyT ist eine Tabellenfunktion, die eine Spalte VARCHAR(100) mit den Namen der Enzyme im Reaktionspfad zurückgibt. Der Name der zurückgegebenen Spalte ist enzyme.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

**Syntax:**

►►—DB2LS.EnzymesByPathwyT—(*pathwayid*)—————►►

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

**Beispiel für das Suchen nach allen Enzymen in einem Reaktionspfad:**

Um nach allen Enzymen in einem Reaktionspfad zu suchen, geben Sie das Argument *pathwayid* in der Funktion an. Beispielsweise möchten Sie nach allen Verbindungen in einem Reaktionspfad 00020 suchen.

Dann lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.EnzymesByPathwyT  
                      ('path:00020')) AS t;
```

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion PathwysByGenesS

Verwenden Sie die Funktion PathwysByGenesS, um nach allen Reaktionspfaden zu suchen, die die von Ihnen angegebenen Gene enthalten.

Die Funktion PathwysByGenesS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Reaktionspfaden zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

```
►► DB2LS.PathwysByGenesS(keggidlist) ◀◀
```

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen nach allen Reaktionspfaden, die spezifische Gene enthalten:

Um nach allen Reaktionspfaden zu suchen, die bestimmte Gene enthalten, geben Sie das Argument *keggidlist* in der Funktion an.

Beispielsweise möchten Sie nach allen Reaktionspfaden suchen, die die E.coli-Gene b0077 und b0078 enthalten.

Dann verwenden Sie folgende Klausel:

```
VALUES CAST(DB2LS.PathwysByGenesS  
          ('eco:b0077 eco:0078')  
          AS VARCHAR(1000));
```

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546



## Benutzerdefinierte Funktion PathwysByGenesT

Verwenden Sie die Funktion PathwysByGenesT, um nach allen Reaktionspfaden zu suchen, die das von Ihnen angegebene Gen enthalten.

Die Funktion PathwysByGenesT ist eine Tabellenfunktion, die eine Spalte VARCHAR(100) mit den Namen der Reaktionspfade für die Gene zurückgibt. Der Name der zurückgegebenen Spalte ist pathway.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

►►—DB2LS.PathwysByGenesT—(*keggidlist*)—◄◄

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen nach allen Reaktionspfaden, die spezifische Gene enthalten:

Um nach allen Reaktionspfaden zu suchen, die bestimmte Gene enthalten, geben Sie das Argument *keggidlist* in der Funktion an.

Beispielsweise möchten Sie nach allen Reaktionspfaden suchen, die die E.coli-Gene b0077 und b0078 enthalten.

Dann lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.PathwysByGenesT
    ('eco:b0077 eco:0078')) AS t;
```

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion PathwysByCompndsS

Verwenden Sie die Funktion PathwysByCompndsS, um nach allen Reaktionspfaden zu suchen, die alle von Ihnen angegebenen Verbindungen enthalten.

Die Funktion PathwysByCompndsS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Verbindungen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

```
►► DB2LS.PathwysByCompndsS(cpdlist) ◀◀
```

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen nach Reaktionspfaden, die alle angegebenen Komponenten enthalten:

Um nach allen Reaktionspfaden zu suchen, die alle Verbindungen einer Liste enthalten, geben Sie das Argument *cpdlist* in der Funktion an.

Beispielsweise möchten Sie nach allen Reaktionspfaden suchen, die die Verbindungen C00033 und C00158 enthalten.

Dann verwenden Sie folgende Klausel:

```
VALUES CAST(DB2LS.PathwysByCompndsS  
           ('cpd:C00033 cpd:C00158')  
           AS VARCHAR(1000));
```

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion PathwysByCompndsT

Verwenden Sie die Funktion PathwysByCompndsT, um nach allen Reaktionspfaden zu suchen, die alle von Ihnen angegebenen Verbindungen enthalten.

Die Funktion PathwysByCompndsT ist eine Tabellenfunktion, die eine Spalte VARCHAR(100) mit den Namen der Reaktionspfade zurückgibt. Der Name der zurückgegebenen Spalte ist pathway.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

```
►► DB2LS.PathwysByCompndsT—(cpdlist)—◄◄
```

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen nach Reaktionspfaden, die alle angegebenen Komponenten enthalten:

Um nach allen Reaktionspfaden zu suchen, die alle Verbindungen einer Liste enthalten, geben Sie das Argument *cpdlist* in der Funktion an.

Beispielsweise möchten Sie nach allen Reaktionspfaden suchen, die die Verbindungen C00033 und C00158 enthalten.

Dann lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.PathwysByCompndsT  
('cpd:C00033 cpd:C00158')) AS t;
```

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion PathwysByEnzymesS

Verwenden Sie die Funktion PathwysByEnzymesS, um nach allen Reaktionspfaden zu suchen, die alle von Ihnen angegebenen Enzyme enthalten.

Die Funktion PathwysByEnzymesS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Reaktionspfaden zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

►►—DB2LS.PathwysByEnzymesS—(*enzymelist*)—►►

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen nach allen Reaktionspfaden, die alle angegebenen Enzyme enthalten:

Um nach allen Reaktionspfaden zu suchen, die alle Enzyme einer Liste enthalten, geben Sie das Argument *enzymelist* in der Funktion an.

Beispielsweise möchten Sie nach allen Reaktionspfaden suchen, die die Enzyme 1.3.99.1 enthalten.

Dann verwenden Sie folgende Klausel:

```
VALUES CAST(DB2LS.PathwysByEnzymesS
           ('ec:1.3.99.1')
           AS VARCHAR(1000));
```

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion PathwysByEnzymesT

Verwenden Sie die Funktion PathwysByEnzymesT, um nach allen Reaktionspfaden zu suchen, die alle von Ihnen angegebenen Enzyme enthalten.

Die Funktion PathwysByEnzymesT ist eine Tabellenfunktion, die eine Spalte VARCHAR(100) mit den Namen der Reaktionspfade zurückgibt. Der Name der zurückgegebenen Spalte ist pathway.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

►►—DB2LS.PathwysByEnzymesT—(*enzymelist*)—►►

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen nach allen Reaktionspfaden, die alle angegebenen Enzyme enthalten:

Um nach allen Reaktionspfaden zu suchen, die alle Enzyme einer Liste enthalten, geben Sie das Argument *enzymelist* in der Funktion an.

Beispielsweise möchten Sie nach allen Reaktionspfaden suchen, die die Enzyme 1.3.99.1 enthalten.

Dann lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.PathwysByEnzymesT
('ec:1.3.99.1')) AS t;
```

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

---

## Sequence Similarity-Datenbankfunktionen

In den nachstehenden Abschnitten werden die benutzerdefinierten Funktionen für die Sequence Similarity-Datenbank (SSDB) erläutert. Diese Abschnitte enthalten Informationen zur Syntax sowie Beispiele der einzelnen Funktionen.

### Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten

Viele der benutzerdefinierten Tabellenfunktionen für die Sequence Similarity-Datenbank (Datenbank für die Sequenzsimilaritätssuche, SSDB) geben einen Satz fixierter Ausgabespalten zurück. Bei Verwendung einer SSDB-Tabellenfunktion können Sie angeben, dass nur eine Untermenge der Spalten zurückgegeben werden sollen. Die Namen und Datentypen der zurückgegebenen Spalten werden in der folgenden Tabelle aufgelistet.

*Tabelle 126. Spalten in der SSDB-Datenbank*

Spaltenname	Spaltendatentyp	Beschreibung
keggid1	VARCHAR (100)	Die Kennung für das Gen, das in der Abfrage angegeben ist.
keggid2	VARCHAR (100)	Die Kennung für das Gen, das von der Abfrage zurückgegeben wird.
swscore	DOUBLE	Der Smith-Waterman-Score zwischen keggid1 und keggid2.
identity	DOUBLE	Der Prozentsatz der Übereinstimmung zwischen keggid1 und keggid2.
overlap	INTEGER	Die Länge der Überlappung zwischen keggid1 und keggid2.
s1_start	INTEGER	Die Startposition der Ausrichtung in keggid1.
s1_end	INTEGER	Die Endposition der Ausrichtung in keggid1.
s2_start	INTEGER	Die Startposition der Ausrichtung in keggid2.
s2_end	INTEGER	Die Endposition der Ausrichtung in keggid2.

Tabelle 126. Spalten in der SSDB-Datenbank (Forts.)

Spaltenname	Spaltentyp	Beschreibung
best1	INTEGER	Die Markierung, die die größte Übereinstimmung von keggid1 zu keggid2 angibt. Ein Wert 1 gibt die eine Beziehung der größten Übereinstimmung von keggid1 zu keggid2 an. Ein Wert 0 gibt an, dass keine Beziehung der größten Übereinstimmung von keggid1 zu keggid2 vorhanden ist.
best2	INTEGER	Die Markierung, die die größte Übereinstimmung von keggid2 zu keggid1 angibt. Ein Wert 1 gibt eine Beziehung der größten Übereinstimmung von keggid2 zu keggid1 an. Ein Wert 0 gibt an, dass keine Beziehung der größten Übereinstimmung von keggid2 zu keggid1 vorhanden ist.
def1	VARCHAR (1000)	Die Definition für keggid1.
def2	VARCHAR (1000)	Die Definition für keggid2.
length1	INTEGER	Die Länge der Aminosäure in keggid1.
length2	INTEGER	Die Länge der Aminosäure in keggid2.

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Referenzen:**

- „Benutzerdefinierte Funktion GenesByPathwyT“ auf Seite 551
- „Benutzerdefinierte Funktion AllNbrsByGeneT“ auf Seite 562
- „Benutzerdefinierte Funktion BstBstNbrsByGeneT“ auf Seite 565
- „Benutzerdefinierte Funktion BestNbrsByGeneT“ auf Seite 567
- „Benutzerdefinierte Funktion RevBestNbrsByGeneT“ auf Seite 569
- „Benutzerdefinierte Funktion ParalogsByGeneT“ auf Seite 571
- „Benutzerdefinierte Funktion BestHmlgsByGenesT“ auf Seite 573
- „Benutzerdefinierte Funktion BstBstHmlgByGenesT“ auf Seite 574

## Benutzerdefinierte Funktion AllNbrsByGeneS

Verwenden Sie die Funktion AllNbrsByGeneS für die Suche nach allen Organismen, die homologe Nachbarn der von Ihnen angegebenen KEGG-Kennung sind.

Anstatt alle Organismen zu durchsuchen, können Sie eine Liste mit Organismen zur Eingrenzung der Suche angeben.

Die Funktion AllNbrsByGeneS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Zielkennungen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

► DB2LS.AllNbrsByGeneS(*keggid*, *threshold* , *orglist*)

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Durchsuchen der gesamten Liste der Organismen:

Um die gesamte Liste der Organismen zu durchsuchen, geben Sie die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise alle homologen Gene nach dem E.coli-Gen mit einem Schwellenwert größer als 200 zu durchsuchen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.AllNbrsByGeneS
           ('eco:b0002', 200)
           AS VARCHAR(1000));
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*.

### Beispiel für das Suchen nach einer spezifischen Gruppe von Organismen:

Um eine Liste mit zu suchenden Organismen zu spezifizieren, geben Sie in der Funktion die Argumente *keggid*, *threshold* und *orglist* an.

Um beispielsweise alle homologen Gene mit einem Schwellenwert größer als 500 zu durchsuchen, wobei nur der Stamm 0157 des E.coli-Gens und alle Stämme des H.influenzae-Gens zurückgegeben werden sollen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.AllNbrsByGeneS
           ('eco:b0002', 500, 'ece hin')
           AS VARCHAR(1000));
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*. Der KEGG-Name für den Stamm 0157 des E.coli-Gens lautet *ece*. Der KEGG-Name für das H.influenzae-Gen lautet *hin*.

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion AllNbrsByGeneT

Verwenden Sie die Funktion AllNbrsByGeneT für die Suche nach allen Organismen, die homologe Nachbarn der von Ihnen angegebenen KEGG-Kennung sind. Anstatt alle Organismen zu durchsuchen, können Sie eine Liste mit Organismen zur Eingrenzung der Suche angeben.

Die Funktion AllNbrsByGeneT ist eine Tabellenfunktion, die eine fixierte Gruppe von Ausgabespalten zurückgibt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

**Syntax:**

► DB2LS.AllNbrsByGeneT(*keggid*, *threshold* [ ,*orglist* ] )

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

**Beispiel für das Durchsuchen der gesamten Liste der Organismen:**

Um die gesamte Liste der Organismen zu durchsuchen, geben Sie die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise alle homologen Gene nach dem E.coli-Gen mit einem Schwellenwert größer als 200 zu durchsuchen, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.AllNbrsByGeneT
  ('eco:b0002', 200)) AS t;
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*.

**Beispiel für das Suchen nach einer spezifischen Gruppe von Organismen:**

Um eine Liste mit zu suchenden Organismen zu spezifizieren, geben Sie in der Funktion die Argumente *keggid*, *threshold* und *orglist* an.

Um beispielsweise alle homologen Gene mit einem Schwellenwert größer als 500 zu durchsuchen, wobei nur der Stamm 0157 des E.coli-Gens und alle Stämme des H.influenzae-Gens zurückgegeben werden sollen, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.AllNbrsByGeneT
  ('eco:b0002', 500, 'ece hin')) AS t;
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*. Der KEGG-Name für den Stamm 0157 des E.coli-Gens lautet *ece*. Der KEGG-Name für das H.influenzae-Gen lautet *hin*.

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545
- „Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten“ auf Seite 560

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547



## Benutzerdefinierte Funktion BstBstNbrsByGeneS

Verwenden Sie die Funktion `BstBstNbrsByGeneS` für die Suche nach den starken besten Nachbarn (Best-Best Neighbors) des Gens in jedem Organismus.

Die Funktion `BstBstNbrsByGeneS` ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Zielkennungen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp `VARCHAR`, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung `SELECT` verwendet werden.

### Syntax:

```
►► DB2LS.BstBstNbrsByGeneS(keggid, threshold [ ,orglist ] )
```

Der Schemaname für diese benutzerdefinierte Funktion ist `DB2LS`.

### Beispiel für das Durchsuchen der gesamten Liste der Organismen:

Um die gesamte Liste der Organismen zu durchsuchen, geben Sie die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise nach allen starken besten Nachbarn (Best-Best Neighbors) des E.coli-Gens mit einem Schwellenwert größer als 200 zu suchen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.BstBstNbrsByGeneS
            ('eco:b0002', 200)
            AS VARCHAR(1000));
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*.

### Beispiel für das Suchen nach einer spezifischen Gruppe von Organismen:

Um eine Liste mit zu suchenden Organismen zu spezifizieren, geben Sie in der Funktion die Argumente *keggid*, *threshold* und *orglist* an.

Um beispielsweise alle starken besten Nachbarn (Best-Best Neighbors) mit einem Schwellenwert größer als 500 zu suchen, wobei nur der Stamm 0157 des E.coli-Gens und alle Stämme des H.influenzae-Gens zurückgegeben werden sollen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.BstBstNbrsByGeneS
            ('eco:b0002', 500, 'ece hin')
            AS VARCHAR(1000));
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*. Der KEGG-Name für den Stamm 0157 des E.coli-Gens lautet *ece*. Der KEGG-Name für das H.influenzae-Gen lautet *hin*.

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

#### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion BstBstNbrsByGeneT

Verwenden Sie die Funktion `BstBstNbrsByGeneT` für die Suche nach den starken besten Nachbarn (Best-Best Neighbors) des Gens in jedem Organismus.

Die Funktion `BstBstNbrsByGeneT` ist eine Tabellenfunktion, die eine fixierte Gruppe von Ausgabespalten zurückgibt.

Diese Funktion kann in einer Anweisung `SELECT` verwendet werden.

#### Syntax:

```
►► DB2LS.BstBstNbrsByGeneT(keggid, threshold, orglist)
```

Der Schemaname für diese benutzerdefinierte Funktion ist `DB2LS`.

#### Beispiel für das Durchsuchen der gesamten Liste der Organismen:

Um die gesamte Liste der Organismen zu durchsuchen, geben Sie die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise nach allen starken besten Nachbarn (Best-Best Neighbors) des E.coli-Gens mit einem Schwellenwert größer als 200 zu suchen, lautet die Anweisung `SELECT` wie folgt:

```
SELECT * FROM TABLE(DB2LS.BstBstNbrsByGeneT  
    ('eco:b0002', 200)) AS t;
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*.

#### Beispiel für das Suchen nach einer spezifischen Gruppe von Organismen:

Um eine Liste mit zu suchenden Organismen zu spezifizieren, geben Sie in der Funktion die Argumente *keggid*, *threshold* und *orglist* an.

Um beispielsweise nach allen starken besten Nachbarn (Best-Best Neighbors) mit einem Schwellenwert größer als 500 zu suchen, wobei nur der Stamm 0157 des E.coli-Gens und alle Stämme des H.influenzae-Gens zurückgegeben werden sollen, lautet die Anweisung `SELECT` wie folgt:

```
SELECT * FROM TABLE(DB2LS.BstBstNbrsByGeneT  
    ('eco:b0002', 500, 'ece hin')) AS t;
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*.

Der KEGG-Name für den Stamm 0157 des E.coli-Gens lautet *ece*. Der KEGG-Name für das H.influenzae-Gen lautet *hin*.

#### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545
- „Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten“ auf Seite 560

#### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

#### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion BestNbrsByGeneS

Verwenden Sie die Funktion BestNbrsByGeneS für die Suche nach den besten Nachbarn (Best Neighbors) des Gens in jedem Organismus.

Die Funktion BestNbrsByGeneS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Zielkennungen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

#### Syntax:

```
►► DB2LS.BestNbrsByGeneS(keggid, threshold [ ,orglist ] )
```

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

#### Beispiel für das Durchsuchen der gesamten Liste der Organismen:

Um die gesamte Liste der Organismen zu durchsuchen, geben Sie die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise nach den besten Nachbarn (Best Neighbors) des E.coli-Gens mit einem Schwellenwert größer als 200 zu suchen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.BestNbrsByGeneS  
           ('eco:b0002', 200)  
           AS VARCHAR(1000));
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*.

#### Beispiel für das Suchen nach einer spezifischen Gruppe von Organismen:

Um eine Liste mit zu suchenden Organismen zu spezifizieren, geben Sie in der Funktion die Argumente *keggid*, *threshold* und *orglist* an.

Um beispielsweise nach den besten Nachbarn (Best Neighbors) mit einem Schwellenwert größer als 500 zu suchen, wobei nur der Stamm 0157 des E.coli-Gens und alle Stämme des H.influenzae-Gens zurückgegeben werden sollen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.BestNbrsByGeneS  
           ('eco:b0002', 500, 'ece hin')  
           AS VARCHAR(1000));
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*. Der KEGG-Name für den Stamm 0157 des E.coli-Gens lautet *ece*. Der KEGG-Name für das H.influenzae-Gen lautet *hin*.

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion BestNbrsByGeneT

Verwenden Sie die Funktion BestNbrsByGeneT für die Suche nach den besten Nachbarn (Best Neighbors) des Gens in jedem Organismus.

Die Funktion BestNbrsByGeneT ist eine Tabellenfunktion, die eine fixierte Gruppe von Ausgabespalten zurückgibt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

**Syntax:**

►► DB2LS.BestNbrsByGeneT(*keggid*, *threshold* [, *orglist*])

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

**Beispiel für das Durchsuchen der gesamten Liste der Organismen:**

Um die gesamte Liste der Organismen zu durchsuchen, geben Sie die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise nach den besten Nachbarn (Best Neighbors) des E.coli-Gens mit einem Schwellenwert größer als 200 zu suchen, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.BestNbrsByGeneT
    ('eco:b0002', 200)) AS t;
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*.

**Beispiel für das Suchen nach einer spezifischen Gruppe von Organismen:**

Um eine Liste mit zu suchenden Organismen zu spezifizieren, geben Sie in der Funktion die Argumente *keggid*, *threshold* und *orglist* an.

Um beispielsweise nach den besten Nachbarn (Best Neighbors) mit einem Schwellenwert größer als 500 zu suchen, wobei nur der Stamm 0157 des E.coli-Gens und alle Stämme des H.influenzae-Gens zurückgegeben werden sollen, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.BestNbrsByGeneT
('eco:b0002', 500, 'ece hin')) AS t;
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*. Der KEGG-Name für den Stamm 0157 des E.coli-Gens lautet *ece*. Der KEGG-Name für das H.influenzae-Gen lautet *hin*.

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545
- „Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten“ auf Seite 560

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion RevBestNbrsByGeneS

Verwenden Sie die Funktion RevBestNbrsByGeneS für die Suche nach den inversen besten Nachbarn (Reverse Best Neighbors) des Gens in jedem Organismus.

Die Funktion RevBestNbrsByGeneS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Zielkennungen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

**Syntax:**

```
DB2LS.RevBestNbrsByGeneS(keggid, threshold, orglist)
```

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

**Beispiel für das Durchsuchen der gesamten Liste der Organismen:**

Um die gesamte Liste der Organismen zu durchsuchen, geben Sie die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise nach den inversen besten Nachbarn (Reverse Best Neighbors) des E.coli-Gens mit einem Schwellenwert größer als 200 zu suchen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.RevBestNbrsByGeneS
('eco:b0002', 200)
AS VARCHAR(1000));
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*.

### Beispiel für das Suchen nach einer spezifischen Gruppe von Organismen:

Um eine Liste mit zu suchenden Organismen zu spezifizieren, geben Sie in der Funktion die Argumente *keggid*, *threshold* und *orglist* an. Um beispielsweise nach den inversen besten Nachbarn (Reverse Best Neighbors) mit einem Schwellenwert größer als 500 zu suchen, wobei nur der Stamm 0157 des E.coli-Gens und alle Stämme des H.influenzae-Gens zurückgegeben werden sollen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.RevBestNbrsByGeneS  
           ('eco:b0002', 500, 'ece hin')  
           AS VARCHAR(1000));
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*. Der KEGG-Name für den Stamm 0157 des E.coli-Gens lautet *ece*. Der KEGG-Name für das H.influenzae-Gen lautet *hin*.

#### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

#### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

#### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion RevBestNbrsByGeneT

Verwenden Sie die Funktion RevBestNbrsByGeneT für die Suche nach den inversen besten Nachbarn (Reverse Best Neighbors) des Gens in jedem Organismus. Die Funktion RevBestNbrsByGeneT ist eine Tabellenfunktion, die eine fixierte Gruppe von Ausgabespalten zurückgibt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

#### Syntax:

```
►► DB2LS.RevBestNbrsByGeneT(keggid, threshold            orglist)
```

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Durchsuchen der gesamten Liste der Organismen:

Um die gesamte Liste der Organismen zu durchsuchen, geben Sie die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise nach den inversen besten Nachbarn (Reverse Best Neighbors) des E.coli-Gens mit einem Schwellenwert größer als 200 zu suchen, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.RevBestNbrsByGeneT  
                    ('eco:b0002', 200)) AS t;
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*.

### Beispiel für das Suchen nach einer spezifischen Gruppe von Organismen:

Um eine Liste mit zu suchenden Organismen zu spezifizieren, geben Sie in der Funktion die Argumente *keggid*, *threshold* und *orglist* an.

Um beispielsweise nach den inversen besten Nachbarn (Reverse Best Neighbors) mit einem Schwellenwert größer als 500 zu suchen, wobei nur der Stamm 0157 des E.coli-Gens und alle Stämme des H.influenzae-Gens zurückgegeben werden sollen, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.RevBestNbrsByGeneT  
('eco:b0002', 500, 'ece hin')) AS t;
```

Der KEGG-Code für das E.coli-Gen lautet *eco*. Der Name des Gens lautet *b0002*. Der KEGG-Name für den Stamm 0157 des E.coli-Gens lautet *ece*. Der KEGG-Name für das H.influenzae-Gen lautet *hin*.

#### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545
- „Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten“ auf Seite 560

#### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

#### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion ParalogByGeneS

Verwenden Sie die Funktion ParalogByGeneS für die Suche nach paralogen Genen in einem Organismus.

Die Funktion ParalogByGeneS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Zielkennungen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

#### Syntax:

```
►►—DB2LS.ParalogByGeneS—(keggid, threshold)—◀◀
```

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

#### Beispiel für das Suchen nach paralogen Genen:

Um nach paralogen Genen in einem Organismus zu suchen, geben Sie nur die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise nach den paralogen Genen im E.coli-Organismus mit einem Schwellenwert größer als 5000 zu suchen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.ParalogsByGeneS  
            ('eco:b0002', 5000)  
            AS VARCHAR(1000));
```

Der KEGG-Datenbankname für den E.coli-Organismus lautet *eco*. Der Name des Organismus lautet *b0002*.

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion ParalogsByGeneT

Verwenden Sie die Funktion ParalogsByGeneT für die Suche nach paralogen Genen in einem Organismus.

Die Funktion ParalogsByGeneT ist eine Tabellenfunktion, die eine fixierte Gruppe von Ausgabespalten zurückgibt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

**Syntax:**

►►—DB2LS.ParalogsByGeneT—(*keggid*, *threshold*)—►►

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

**Beispiel für das Suchen nach paralogen Genen:**

Um nach paralogen Genen in einem Organismus zu suchen, geben Sie die Argumente *keggid* und *threshold* in der Funktion an.

Um beispielsweise nach den paralogen Genen im E.coli-Organismus mit einem Schwellenwert größer als 5000 zu suchen, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.ParalogsByGeneT  
                    ('eco:b0002', 5000)) AS t;
```

Der KEGG-Datenbankname für den E.coli-Organismus lautet *eco*. Der Name des Organismus lautet *b0002*.

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545
- „Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten“ auf Seite 560



#### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

#### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion BestHmlgsByGenesS

Verwenden Sie die Funktion BestHmlgsByGenesS für die Suche nach den besten homologen Nachbarn eines Organismus innerhalb einer von Ihnen angegebenen Liste mit Genen.

Die Funktion BestHmlgsByGenesS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Zielkennungen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

#### Syntax:

► DB2LS.BestHmlgsByGenesS(*keggorg*, *kegidlist*) ◀

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

Um eine bessere Leistung zu erzielen, verwenden Sie die Funktion BstBstHmlgByGenesS.

#### Beispiel für das Suchen nach den besten homologen Nachbarn:

Um nach den besten homologen Nachbarn eines Organismus innerhalb einer Liste mit Genen zu suchen, geben Sie in der Funktion die Argumente *keggorg* und *kegidlist* an.

Um beispielsweise nach dem H.influenzae-Organismus in der Liste der E.coli-Gene b0002, b0003, b0004 und b0005 zu suchen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.BestHmlgsByGenesS
            ('hin', 'eco:b0002 eco:b0003 eco:b0004 eco:b0005')
            VARCHAR(1000));
```

#### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

#### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

#### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion BestHmlgsByGenesT

Verwenden Sie die Funktion BestHmlgsByGenesT für die Suche nach den besten homologen Nachbarn eines Organismus innerhalb einer von Ihnen angegebenen Liste mit Genen.

Die Funktion BestHmlgsByGenesT ist eine Tabellenfunktion, die eine fixierte Gruppe von Ausgabespalten zurückgibt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

►►—DB2LS.BestHmlgsByGenesT—(*keggorg*, *keggidlist*)—►►

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

Um eine bessere Leistung zu erzielen, verwenden Sie die Funktion BstBstHmlgByGenesT.

### Beispiel für das Suchen nach den besten homologen Nachbarn:

Um nach den besten homologen Nachbarn eines Organismus innerhalb einer Liste mit Genen zu suchen, geben Sie in der Funktion die Argumente *keggorg* und *keggidlist* an.

Um beispielsweise nach dem H.influenzae-Organismus in der Liste der E.coli-Gene b0002, b0003, b0004 und b0005 zu suchen, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.BestHmlgsByGenesT
    ('hin', 'eco:b0002 eco:b0003 eco:b0004 eco:b0005')) AS t;
```

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545
- „Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten“ auf Seite 560

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion BstBstHmlgByGenesS

Verwenden Sie die Funktion BstBstHmlgByGenesS für die Suche nach den starken besten homologen Nachbarn (Best-Best Homologous Neighbors) eines Organismus innerhalb einer Liste mit Genen.

Die Funktion BstBstHmlgByGenesS ist eine Skalarfunktion, die eine durch Leerzeichen unterteilte Liste mit Zielkennungen zurückgibt. Diese Liste ist eine Zeichenfolge vom Datentyp VARCHAR, deren tatsächliche Länge nicht mehr als 32767 Byte beträgt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

**Syntax:**

►►—DB2LS.BstBstHmlgByGenesS—(*keggorg*, *keggidlist*)—►►

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

**Beispiel für das Suchen nach ausschließlich den starken besten Nachbarn (Best-Best Neighbors):**

Um ausschließlich nach den starken besten homologen Nachbarn (Best-Best Homologous Neighbors) eines Organismus innerhalb einer Liste mit Genen zu suchen, geben Sie in der Funktion die Argumente *keggorg* und *keggidlist* an.

Um beispielsweise nach dem H.influenzae-Organismus in der Liste der E.coli-Gene b0002, b0003, b0004 und b0005 zu suchen, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.BstBstHmlgByGenesS
            ('hin', 'eco:b0002 eco:b0003 eco:b0004 eco:b0005')
            AS VARCHAR(1000));
```

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion BstBstHmlgByGenesT

Verwenden Sie die Funktion BstBstHmlgByGenesT für die Suche nach den starken besten homologen Nachbarn (Best-Best Homologous Neighbors) eines Organismus innerhalb einer Liste mit Genen.

Die Funktion BstBstHmlgByGenesT ist eine Tabellenfunktion, die eine fixierte Gruppe von Ausgabespalten zurückgibt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

**Syntax:**

►►—DB2LS.BstBstHmlgByGenesT—(*keggorg*, *keggidlist*)—►►

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen nach den besten der besten Nachbarn:

Um ausschließlich nach den starken besten Nachbarn (Best-Best Neighbors) eines Organismus innerhalb einer Liste mit Genen zu suchen, geben Sie in der Funktion die Argumente *keggorg* und *keggidlist* an.

Um beispielsweise nach dem H.influenzae-Organismus in der Liste der E.coli-Gene b0002, b0003, b0004 und b0005 zu suchen, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.BstBstHmlgByGenesT
    ('hin', 'eco:b0002 eco:b0003 eco:b0004 eco:b0005')) AS t;
```

#### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545
- „Von SSDB-Datenbankabfragen (Tabellenfunktionen) zurückgegebene Spalten“ auf Seite 560

#### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

#### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion ScoreBetweenGenesS

Verwenden Sie die Funktion *ScoreBetweenGenesS* für die Ermittlung des Smith-Waterman-Score zwischen zwei Genen.

Die Funktion *ScoreBetweenGenesS* ist eine Skalarfunktion, die einen einzelnen Wert vom Datentyp DOUBLE zurückgibt.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

#### Syntax:

```
►►—DB2LS.ScoreBetweenGenesS—(keggid1, keggid2)—————◄◄
```

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

#### Beispiel für das Ermitteln des Smith-Waterman-Score zwischen zwei Genen:

Um den Smith-Waterman-Score zwischen zwei Genen zu ermitteln, geben Sie die *keggid* für jedes Gen an.

Um beispielsweise den Smith-Waterman-Score zwischen den E.Coli-Genen b0002 und b3940 zu ermitteln, lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.ScoreBetweenGenesS
    ('eco:b0002', 'eco:b3940')
    AS DOUBLE);
```

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion `DefinitionsByGeneS`

Verwenden Sie die Funktion `DefinitionsByGeneS` für die Suche nach der Definition eines Gens.

Die Funktion `DefinitionsByGeneS` ist eine Skalarfunktion, die einen einzelnen Wert vom Datentyp `VARCHAR(1000)` zurückgibt.

Diese Funktion kann in einer Anweisung `SELECT` verwendet werden.

**Syntax:**

```
►► DB2LS.DefinitionsByGeneS(keggid) ◀◀
```

Der Schemaname für diese benutzerdefinierte Funktion ist `DB2LS`.

**Beispiel für das Suchen nach der Definition eines Gens:**

Um nach der Definition eines Gens zu suchen, geben Sie die *keggid* an.

Für die Rückgabe der Definition für das E.coli-Gen `b0002` beispielsweise lautet die zu verwendende Klausel wie folgt:

```
VALUES CAST(DB2LS.DefinitionsByGeneS  
           ('eco:b0002')) AS t;  
AS VARCHAR(1000));
```

**Zugehörige Konzepte:**

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

**Zugehörige Tasks:**

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

**Zugehörige Referenzen:**

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

## Benutzerdefinierte Funktion GenesByMotifsT

Verwenden Sie die Funktion GenesByMotifsT für die Suche nach Genen, die alle Motive aus einer von Ihnen angegebenen Liste mit Motiven enthalten.

Die Funktion GenesByMotifsT ist eine Tabellenfunktion, die eine Tabelle zurückgibt. Die Tabelle enthält die ID *keggid* VARCHAR(100) und die Definition *definition* VARCHAR(1000) für jedes Gen.

Diese Funktion kann in einer Anweisung SELECT verwendet werden.

### Syntax:

```
►► DB2LS.GenesByMotifsT(midlist) ◀◀
```

Der Schemaname für diese benutzerdefinierte Funktion ist DB2LS.

### Beispiel für das Suchen nach Genen, die alle Motive einer Liste enthalten:

Um nach Genen zu suchen, die alle Motive einer Liste enthalten, geben Sie das Argument *midlist* an.

Für die Suche nach Genen beispielsweise, die sowohl das Motiv *Pfam* 'DnaJ' als auch das Motiv *Prosite* 'DNAJ\_2' enthalten, lautet die Anweisung SELECT wie folgt:

```
SELECT * FROM TABLE(DB2LS.GenesByMotifsT  
    ('pf:DnaJ ps:DNAJ_2')) AS t;
```

### Zugehörige Konzepte:

- „Benutzerdefinierte KEGG-Funktionen - Übersicht“ auf Seite 545

### Zugehörige Tasks:

- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

### Zugehörige Referenzen:

- „Funktionsargumente für benutzerdefinierte KEGG-Funktionen“ auf Seite 547
- „Benutzerdefinierte KEGG-Funktionen nach Funktionskategorie“ auf Seite 546

---

## Inaktivieren der benutzerdefinierten KEGG-Funktionen

Wenn Sie die benutzerdefinierten KEGG-Funktionen nicht länger verwenden möchten, können Sie die Funktionen vorübergehend inaktivieren oder permanent aus ihrer zusammengeschlossenen Datenbank entfernen.

Falls Sie die benutzerdefinierten KEGG-Funktionen inaktivieren, können Sie diese durch Registrieren in der zusammengeschlossenen Datenbank erneut aktivieren.

Um die Funktionen permanent aus der zusammengeschlossenen Datenbank zu entfernen, müssen Sie sie deinstallieren.

### Vorgehensweise:

Um die benutzerdefinierten KEGG-Funktionen zu inaktivieren, führen Sie den Befehl 'disable\_KEGGFunctions' aus.

- Auf Servern mit zusammengeschlossenen Datenbanken, die unter Windows ausgeführt werden, befindet sich dieser Befehl im Verzeichnis `sqllib\bin`.
- Auf Servern mit zusammengeschlossenen Datenbanken, die unter UNIX ausgeführt werden, befindet sich dieser Befehl im Verzeichnis `sqllib\bin`.

Syntax:

```
disable_KEGGFunctions -n dbName -u benutzerID -p kennwort
```

#### **dbname**

Der Name der zusammengeschlossenen Datenbank, für die Sie die Funktionen inaktivieren möchten.

#### **benutzerID**

Eine gültige Benutzer-ID für die zusammengeschlossene Datenbank.

#### **kennwort**

Ein gültiges Kennwort für die Benutzer-ID.

### Beispiel für das Inaktivieren von benutzerdefinierten KEGG-Funktionen:

Das folgende Beispiel zeigt die Ausgabe, die beim Absetzen des Befehls 'disable\_KEGGFunctions' zurückgegeben wird:

```
C:>disable_KEGGFunctions -n federateddb -u db2admin -p db2admin
```

```
(37) KEGG Functions were found
      -- Drop KEGG Functions ...
      Drop KEGG Functions Successfully.
```

```
*** Please allow a few seconds to clean up the system .....
```

### Zugehörige Tasks:

- „Entfernen von relationalen Wrappern, nicht relationalen Wrappern und benutzerdefinierten Life Sciences-Funktionen (Windows)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Entfernen von Wrappern, benutzerdefinierten Funktionen und Wrapper Development Kits (UNIX)“ in *IBM DB2 Information Integrator Installationshandbuch für Linux, UNIX und Windows*
- „Registrieren der benutzerdefinierten KEGG-Funktionen“ auf Seite 548

---

## Teil 5. Referenz





---

## Kapitel 27. Unterstützte Datentypen für nicht relationale Datenquellen

Die nachstehenden Abschnitte enthalten eine Liste der Datentypen, die von den nicht relationalen Wrappern unterstützt werden.

---

### Unterstützte Datentypen für nicht relationale Datenquellen

Bei den meisten nicht relationalen Datenquellen müssen Sie die Spalteninformationen (einschließlich Datentyp) angeben, wenn Sie die Kurznamen für den Zugriff auf die betreffenden Datenquellen erstellen.

Einige der nicht relationalen Wrapper erstellen alle Spalten, die für den Zugriff auf eine Datenquelle erforderlich sind. Diese Spalten werden als *fixierte Spalten* bezeichnet. Bei anderen Wrappern können Sie einige oder alle Datentypen für die Spalten in der Anweisung CREATE NICKNAME angeben.

Die folgenden Abschnitte enthalten eine Liste der Wrapper, für die Sie die Datentypen angeben können, sowie eine Liste der Datentypen, die von den Wrappern jeweils unterstützt werden.

#### Vom BioRS-Wrapper unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom BioRS-Wrapper unterstützt werden.

*Tabelle 127. Zuordnungen zwischen BioRS-Datentypen und DB2-Datentypen*

BioRS-Datentypen	DB2-Datentypen
AUTHOR	CHARACTER, CLOB, VARCHAR
DATE	CHARACTER, CLOB, VARCHAR
NUMBER	CHARACTER, CLOB, VARCHAR
REFERENCE	CHARACTER, CLOB, VARCHAR
TEXT	CHARACTER, CLOB, VARCHAR

Die maximal zulässige Länge für den Datentyp CLOB ist 5 Megabyte.

#### Vom BLAST-Wrapper unterstützte Datentypen

Einige Datentypen werden automatisch für die fixierten Spalten gesetzt, die der BLAST-Wrapper erstellt.

Für die Definitionszeilenfelder können Sie einen Datentyp zuordnen, wenn Sie einen Kurznamen erstellen. Sind die Daten in der Definitionszeilenspalte (definition line) mit dem Datentyp der lokalen Spalte nicht kompatibel, erhalten Sie eine Fehlermeldung. Wenn Sie beispielsweise eine Definitionszeilenspalte vom Typ INTEGER definieren und die Spalte Werte enthält, die nicht numerisch sind, wird eine Fehlermeldung zurückgegeben.

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom BLAST-Wrapper unterstützt werden.

Tabelle 128. Zuordnungen zwischen BLAST-Datentypen und DB2-Datentypen

BLAST-Datentypen	DB2-Datentypen
Definitionszeile (definition line)	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB ist 5 Megabyte.
Definitionszeile (definition line)	DOUBLE
Definitionszeile (definition line)	FLOAT
Definitionszeile (definition line)	INTEGER
Definitionszeile (definition line)	VARCHAR

## Vom Documentum-Wrapper unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom Documentum-Wrapper unterstützt werden.

Tabelle 129. Zuordnungen zwischen Documentum-Datentypen und DB2-Datentypen

Documentum-Datentypen	DB2-Datentypen
DOUBLE	DOUBLE, FLOAT, INTEGER, SMALLINT
ID	CHARACTER (16)
INTEGER	DOUBLE, FLOAT, INTEGER, SMALLINT
STRING (bis zu 255 Zeichen)	CHAR, VARCHAR
TIME	CHAR, DATE, TIMESTAMP, VARCHAR

## Vom Entrez-Wrapper unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom Entrez-Wrapper unterstützt werden.

Tabelle 130. Zuordnungen zwischen Entrez-Datentypen und DB2-Datentypen

Entrez-Datentypen	DB2-Datentypen
Zeichen (character)	CHARACTER
Zeichen (character)	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB ist 5 Megabyte.
Datum (date)	DATE
Zahl (number)	DECIMAL
Zahl (number)	DOUBLE
Ganze Zahl (integer)	INTEGER
Zahl (number)	REAL
Ganze Zahl (integer)	SMALLINT
Zeit (time)	TIMESTAMP
Zeichen (character)	VARCHAR

## Vom Excel-Wrapper unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom Excel-Wrapper unterstützt werden.

*Tabelle 131. Zuordnungen zwischen Excel-Datentypen und DB2-Datentypen*

Excel-Datentypen	DB2-Datentypen
Datum (date)	DATE
Zahl (number)	DOUBLE
Zahl (number)	FLOAT (n), wobei $n \geq 25$ und $\leq 53$ ist
Ganze Zahl (integer)	INTEGER
Zeichen (character)	VARCHAR

## Vom Extended Search-Wrapper unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom Extended Search-Wrapper unterstützt werden.

*Tabelle 132. Zuordnungen zwischen Extended Search-Datentypen und DB2-Datentypen*

Extended Search-Datentypen	DB2-Datentypen
Datum (date)	DATE
Double (Datentyp)	DOUBLE
Ganze Zahl (integer)	INTEGER
Zeichenfolge (string)	VARCHAR

## Vom HMMER-Wrapper unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom HMMER-Wrapper unterstützt werden.

*Tabelle 133. Zuordnungen zwischen HMMER-Datentypen und DB2-Datentypen*

HMMER-Datentypen	DB2-Datentypen
Zeichen (character)	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB ist 5 Megabyte.
Zeichen (character)	DOUBLE
Zeichen (character)	FLOAT
Zeichen (character)	INTEGER
Zeichen (character)	VARCHAR

## Vom Wrapper für Dateien mit Tabellenstruktur unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom Wrapper für Dateien mit Tabellenstruktur unterstützt werden.

*Tabelle 134. Zuordnungen zwischen Datentypen von Dateien mit Tabellenstruktur und DB2-Datentypen*

Datentypen von Dateien mit Tabellenstruktur	DB2-Datentypen
Zeichen (character)	CHARACTER
Zeichen (character)	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB ist 5 Megabyte.
Zahl (number)	DECIMAL
Zahl (number)	DOUBLE
Zahl (number)	FLOAT
Ganze Zahl (integer)	INTEGER
Zahl (number)	REAL
Ganze Zahl (integer)	SMALLINT
Zeichen (character)	VARCHAR

## Vom Web-Services-Wrapper unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom Web-Services-Wrapper unterstützt werden. Der Web-Services-Wrapper verwendet XML-Datentypen.

*Tabelle 135. Zuordnungen zwischen XML-Datentypen und DB2-Datentypen für den Web-Services-Wrapper*

XML-Datentypen	DB2-Datentypen
Zeichen (character)	CHARACTER
Zeichen (character)	CHARACTER FOR BIT DATA
Zeichen (character)	CLOB
Datum (date)	DATE
Zahl (number)	DECIMAL
Zahl (number)	DOUBLE
Zahl (number)	FLOAT
Ganze Zahl (integer)	INTEGER
Zahl (number)	REAL
Ganze Zahl (integer)	SMALLINT
Zeichen (character)	VARCHAR
Zeichen (character)	VARCHAR FOR BIT DATA

## Vom WebSphere Business Integration-Wrapper unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom WebSphere Business Integration-Wrapper unterstützt werden. Der WebSphere Business Integration-Wrapper verwendet XML-Datentypen.

*Tabelle 136. Zuordnungen zwischen XML-Datentypen und DB2-Datentypen für den WebSphere Business Integration-Wrapper*

XML-Datentypen	DB2-Datentypen
Zeichen (character)	CHARACTER
Zeichen (character)	CHARACTER FOR BIT DATA
Zeichen (character)	CLOB
Datum (date)	DATE
Zahl (number)	DECIMAL
Zahl (number)	DOUBLE
Zahl (number)	FLOAT
Ganze Zahl (integer)	INTEGER
Zahl (number)	REAL
Ganze Zahl (integer)	SMALLINT
Zeichen (character)	VARCHAR
Zeichen (character)	VARCHAR FOR BIT DATA

## Vom XML-Wrapper unterstützte Datentypen

Die folgende Tabelle enthält eine Liste der DB2-Datentypen, die vom XML-Wrapper unterstützt werden.

*Tabelle 137. Zuordnungen zwischen XML-Datentypen und DB2-Datentypen für den XML-Wrapper*

XML-Datentypen	DB2-Datentypen
Zeichen (character)	CHARACTER
Zeichen (character)	CHARACTER FOR BIT DATA
Zeichen (character)	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB ist 5 Megabyte.
Datum (date)	DATE
Zahl (number)	DECIMAL
Zahl (number)	DOUBLE
Zahl (number)	FLOAT
Ganze Zahl (integer)	INTEGER
Zahl (number)	REAL
Ganze Zahl (integer)	SMALLINT
Zeichen (character)	VARCHAR
Zeichen (character)	VARCHAR FOR BIT DATA

|  
|  
|  
|  
|  
|  
|

**Zugehörige Konzepte:**

- „Datentypzuordnungen in einem System zusammengeschlossener Datenbanken“ in *Systeme zusammengeschlossener Datenbanken*
- „Datentypzuordnungen und der globale Katalog zusammengeschlossener Datenbanken“ in *Systeme zusammengeschlossener Datenbanken*
- „Datentypzuordnungen für nicht relationale Datenquellen“ in *Systeme zusammengeschlossener Datenbanken*

---

## Kapitel 28. Ändern von Kurznamen

In diesem Kapitel wird erläutert, wie zuvor registrierte Kurznamen geändert werden.

---

### Ändern eines Kurznamens

Kurznamen sind Kennungen, die zum Verweisen auf ein Objekt verwendet werden, auf das Sie in einer Datenquelle zugreifen möchten.

Sie können einen Kurznamen ändern, um folgende Aktionen durchzuführen:

- Ändern der lokalen Spaltennamen für die Spalten des Datenquellenobjekts
- Ändern der lokalen Datentypen für die Spalten des Datenquellenobjekts
- Hinzufügen, Definieren oder Löschen von Kurznamen- und Spaltenoptionen
- Hinzufügen oder Löschen eines Primärschlüssels
- Hinzufügen oder Löschen einer oder mehrerer eindeutiger oder referenzieller Integritätsbedingungen sowie die Prüfungen auf Integritätsbedingung
- Ändern einer oder mehrerer Attribute für referenzielle Integritätsbedingung, Prüfung auf Integritätsbedingung oder Integritätsbedingung für funktionale Abhängigkeit

#### Voraussetzungen:

Die Berechtigungs-ID der Anweisung muss zumindest über eines der folgenden Zugriffsrechte verfügen:

- Berechtigung SYSADM oder DBADM
- Zugriffsrecht ALTER für den Kurznamen, der in der Anweisung angegeben wurde
- Zugriffsrecht CONTROL für den Kurznamen, der in der Anweisung angegeben wurde
- Zugriffsrecht ALTERIN für das Schema, wenn der Schemaname des Kurznamens vorhanden ist
- Definierender Benutzer des Kurznamens, der in der Spalte DEFINER der Katalogsicht für den Kurznamen aufgezeichnet wurde

#### Einschränkungen:

Informationen zu den Einschränkungen beim Ändern von Kurznamen finden Sie im entsprechenden Abschnitt.

#### Vorgehensweise:

Sie können einen Kurznamen von der DB2-Steuerzentrale oder der DB2-Befehlszeile aus ändern.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Wählen Sie den Ordner **Kurznamen** aus.



2. Klicken Sie mit der rechten Maustaste auf den Kurznamen, den Sie ändern möchten, und klicken Sie auf **Ändern**. Das Notizbuch **Kurznamen ändern** wird geöffnet.
3. Ändern Sie auf der Seite **Kurznamen** die lokalen Spaltennamen, die lokalen Datentypen oder die Spaltenoptionen für die Spalten, die im globalen Katalog gespeichert sind.
4. Definieren Sie auf der Seite **Schlüssel** die referenziellen Integritätsbedingungen für den Kurznamen. Sie können eine über Primärschlüssel, eindeutige Schlüssel oder Fremdschlüssel definierte Integritätsbedingung festlegen.
5. Legen Sie auf der Seite **Prüfung auf Integritätsbedingung** die Prüfungen auf Integritätsbedingung bzw. die Integritätsbedingungen der funktionalen Abhängigkeiten für den Kurznamen fest.
6. Definieren Sie auf der Seite **Einstellungen** die Kurznamenoptionen für den Kurznamen.
7. Klicken Sie auf **OK**, um den Kurznamen zu ändern, und schließen Sie das Notizbuch.  
 Einige Kurznamenoptionen sind erforderlich und können nicht gelöscht werden. Andere Kurznamenoptionen können nicht hinzugefügt werden, wenn bereits bestimmte Kurznamenoptionen definiert wurden. Eine Liste der Beschreibungen zu den Optionen finden Sie unter den Kurznamenoptionen für Systeme zusammengeschlossener Datenbanken und unter den Kurznamenspaltenoptionen für Systeme zusammengeschlossener Datenbanken.

Geben Sie die Anweisung ALTER NICKNAME ein, um diese Task von der DB2-Befehlszeile aus auszuführen; dabei müssen die entsprechenden Parameter definiert sein.

Wenn die Struktur bzw. der Inhalt des Datenquellenobjekts wesentlich geändert wird, müssen die Kurznamenstatistiken aktualisiert werden. Derartige Änderungen beinhalten das Hinzufügen und Entfernen mehrerer Zeilen.

#### **Zugehörige Konzepte:**

- „Informative Integritätsbedingungen für Kurznamen“ in *Systeme zusammengeschlossener Datenbanken*
- „Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht“ in *Systeme zusammengeschlossener Datenbanken*

#### **Zugehörige Tasks:**

- „Ändern von Kurznamenoptionen“ auf Seite 592
- „Ändern eines lokalen Typs für ein Datenquellenobjekt“ auf Seite 595
- „Ändern der Spaltennamen von Kurznamen“ auf Seite 590
- „Ändern der Spaltenoptionen von Kurznamen“ auf Seite 593

#### **Zugehörige Referenzen:**

- „Einschränkungen für das Ändern von Kurznamen“ auf Seite 589
- Anhang F, „Kurznamenoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 661
- Anhang G, „Kurznamenspaltenoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 671
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*

---

## Einschränkungen für das Ändern von Kurznamen

Für das Ändern von Kurznamen gelten folgende Einschränkungen.

### Spaltennamen

Die Anweisung ALTER NICKNAME kann nicht zur Änderung von Spaltennamen für folgende Datenquellen verwendet werden. Sie müssen den Kurznamen löschen und ihn mit den korrekten Spaltennamen erneut erstellen.

- BLAST
- Documentum
- HMMER

### Spaltenoptionen

Wenn eine der folgenden Optionen für eine Spalte gesetzt ist, können keine weiteren Optionen zu dieser Spalte hinzugefügt werden:

- SOAPACTIONCOLUMN
- URLCOLUMN
- PRIMARY\_KEY
- FOREIGN\_KEY

Für BioRS

- Wird der Elementname einer Spalte mit Hilfe der Option ELEMENT\_NAME geändert, wird der neue Name nicht auf Richtigkeit überprüft. Eine inkorrekte Option kann zu Fehlern führen, wenn in einer Abfrage auf die Spalte verwiesen wird.
- Wenn an der Spaltenoption IS\_INDEXED Änderungen vorgenommen werden, werden diese Änderungen nicht mit dem BioRS-Server überprüft. Eine inkorrekte Option kann zu Fehlern führen, wenn in einer Abfrage auf die Spalte verwiesen wird.

### Datentypen

- Wird der Datentyp einer Spalte geändert, muss der neue Datentyp mit dem Datentyp der entsprechenden Spalte bzw. des entsprechenden Elements der Datenquelle kompatibel sein. Das Ändern des lokalen Datentyps in einen Datentyp, der mit dem fernen Datentyp nicht kompatibel ist, kann unvorhersehbare Fehler verursachen.
- Der *lokale\_datentyp* darf nicht LONG VARCHAR, LONG VARGRAPHIC, DATALINK oder ein benutzerdefinierter Datentyp sein.
- Der *datentyp\_der\_datenquelle* darf kein benutzerdefinierter Typ sein.
- Für einige der nicht relationalen Datenquellen können vorhandene lokale Typen nicht überschrieben werden bzw. keine neuen lokalen Typen erstellt werden. Weitere Informationen zu dieser Einschränkung finden Sie in der Dokumentation für den jeweiligen Datenquellenwrapper.
- Wird die lokale Spezifikation eines Spaltendatentyps geändert, macht der Datenbankmanager der zusammengeschlossenen Datenbank alle Statistikdaten ungültig, die für diese Spalte erfasst wurden (z. B. HIGH2KEY und LOW2KEY).
- Der lokale Typ wird für das spezifische Datenquellenobjekt definiert, wenn mit diesem Kurznamen darauf zugegriffen wird. Dasselbe Datenquellenobjekt kann über einen weiteren Kurznamen verfügen, der die standardmäßige Datentypzuordnung verwendet.

## Indizes

Die Anweisung ALTER NICKNAME kann nicht zur Registrierung eines neuen Datenquellenindex in der zusammengesetzten Datenbank verwendet werden. Verwenden Sie zum Erstellen einer Indexspezifikation die Anweisung CREATE INDEX mit der Klausel SPECIFICATION ONLY.

## Parameter LOCAL NAME und LOCAL TYPE

- Die Anweisung ALTER NICKNAME kann in folgenden Fällen nicht zum Ändern der lokalen Namen oder Datentypen für die Spalten in dem Kurznamen verwendet werden:
  - Der Kurzname wird in einer Sicht, SQL-Methode oder SQL-Funktion verwendet
  - Für den Kurznamen ist eine Informationsintegritätsbedingung definiert
- Falls in der Anweisung ALTER NICKNAME auch die Angabe des Parameters LOCAL NAME und/oder LOCAL TYPE erforderlich ist, muss die Klausel 'federated\_column\_options' zuletzt angegeben werden.

## Kurznamen

Die Anweisung ALTER NICKNAME kann nicht zur Änderung des Namens der BioRS-Datenbank verwendet werden, auf die ein BioRS-Kurzname verweist bzw. die in diesem Kurznamen verwendet wird. Ändert sich der Name einer BioRS-Datenbank, müssen Sie den Kurznamen löschen und erneut erstellen.

## Arbeitseinheiten

Der Server mit zusammengesetzten Datenbanken kann eine Anweisung ALTER NICKNAME innerhalb einer bestimmten Arbeitseinheit nicht verarbeiten, wenn eine der folgenden Bedingungen zutrifft:

- Der Kurzname, auf den in der Anweisung ALTER NICKNAME verwiesen wird, hat einen Cursor in derselben Arbeitseinheit geöffnet.
- Für den Kurznamen, auf den in der Anweisung ALTER NICKNAME verwiesen wird, erfolgt das Absetzen einer Einfüge-, Lösch- oder Aktualisierungsoperation in derselben Arbeitseinheit.
- Bei nicht relationalen Datenquellen: Die Anweisung ALTER NICKNAME verweist auf einen Kurznamen, auf den in derselben Arbeitseinheit durch eine Anweisung SELECT verwiesen wird.

## Zugehörige Tasks:

- „Ändern von Kurznamenoptionen“ auf Seite 592
- „Ändern eines lokalen Typs für ein Datenquellenobjekt“ auf Seite 595
- „Ändern eines Kurznamens“ auf Seite 587
- „Ändern der Spaltennamen von Kurznamen“ auf Seite 590
- „Ändern der Spaltenoptionen von Kurznamen“ auf Seite 593

---

## Ändern der Spaltennamen von Kurznamen

Wenn Sie einen Kurznamen erstellen, werden die Spaltennamen, die dem Datenquellenobjekt zugeordnet sind, in der zusammengesetzten Datenbank gespeichert. Bei einigen Datenquellen gibt der Wrapper die Spaltennamen an. Für andere Datenquellen müssen Sie bei der Erstellung des Kurznamens die Spaltennamen angeben.

Sie können einen Kurznamen ändern, um die Spaltennamen zu ändern.

### Voraussetzungen:

Die Berechtigungs-ID, mit der die Anweisung eingegeben wird, muss über mindestens eines der folgenden Zugriffsrechte verfügen:

- Berechtigung SYSADM oder DBADM
- Zugriffsrecht ALTER für den Kurznamen, der in der Anweisung angegeben wurde
- Zugriffsrecht CONTROL für den Kurznamen, der in der Anweisung angegeben wurde
- Zugriffsrecht ALTERIN für das Schema, wenn der Schemaname des Kurznamens vorhanden ist
- Definierender Benutzer des Kurznamens, der in der Spalte DEFINER der Katalogsicht für den Kurznamen aufgezeichnet wurde

### Einschränkungen:

Informationen zu den Einschränkungen beim Ändern von Kurznamen finden Sie im entsprechenden Abschnitt.

### Vorgehensweise:

Sie können Spaltennamen von der DB2-Steuerzentrale oder der DB2-Befehlszeile aus ändern.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Wählen Sie den Ordner **Kurznamen** aus.
2. Klicken Sie mit der rechten Maustaste auf den Kurznamen, den Sie ändern möchten, und klicken Sie auf **Ändern**. Das Notizbuch **Kurznamen ändern** wird geöffnet.
3. Wählen Sie auf der Seite **Kurznamen** die Spalte aus, die Sie ändern möchten, und klicken Sie auf **Ändern**. Das Fenster **Spalte ändern** wird geöffnet.
4. Geben Sie den Spaltennamen ein.
5. Klicken Sie auf **OK**, um den Spaltennamen zu ändern, und schließen Sie das Fenster.
6. Klicken Sie auf **OK**, um den Kurznamen zu ändern, und schließen Sie das Notizbuch.

Geben Sie die Anweisung ALTER NICKNAME ein, um diese Task von der DB2-Befehlszeile aus auszuführen:

```
ALTER NICKNAME nickname  
ALTER COLUMN current_name  
LOCAL NAME new_name
```

### Beispiel: Ändern des lokalen Namens einer Kurznamenspalte:

Der Kurzname Z\_EMPLOYEES für eine Tabelle von DB2 UDB für z/OS beinhaltet eine Spalte mit dem Namen EMPNO. Geben Sie die folgende Anweisung ein, um den Kurznamen zu ändern, sodass der lokale Spaltenname, mit dem Benutzer arbeiten, *Employee\_Number* an Stelle von *EMPNO* lautet:

```
ALTER NICKNAME Z_EMPLOYEES ALTER COLUMN EMPNO  
LOCAL NAME "Employee_Number"
```

#### Zugehörige Tasks:

- „Ändern eines Kurznamens“ auf Seite 587

#### Zugehörige Referenzen:

- „Einschränkungen für das Ändern von Kurznamen“ auf Seite 589
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*

---

## Ändern von Kurznamenoptionen

Kurznamenoptionen sind Parameter, die Sie für den Kurznamen beim Ausgeben der Anweisungen CREATE NICKNAME und ALTER NICKNAME angeben.

Sie können Kurznamenoptionen mit Hilfe der Anweisung ALTER NICKNAME hinzufügen, setzen oder löschen.

#### Voraussetzungen:

Die Berechtigungs-ID für die Ausgabe der Anweisung muss mindestens eines der folgenden Zugriffsrechte enthalten:

- Berechtigung SYSADM oder DBADM
- Zugriffsrecht ALTER für den in der Anweisung angegebenen Kurznamen
- Zugriffsrecht CONTROL für den in der Anweisung angegebenen Kurznamen
- Zugriffsrecht ALTERIN für das Schema, falls der Schemaname des Kurznamens vorhanden ist
- Definierender Benutzer des Kurznamens, wie in der Spalte DEFINER der Katalogsicht für den Kurznamen eingetragen

#### Einschränkungen:

Siehe hierzu den Abschnitt zu Einschränkungen beim Ändern von Kurznamen.

#### Vorgehensweise:

Sie können Spaltennamen von der DB2-Steuerzentrale aus oder über die DB2-Befehlszeile ändern.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Wählen Sie den Ordner **Kurznamen** aus.
2. Klicken Sie mit der rechten Maustaste den zu ändernden Kurznamen an, und klicken Sie **Ändern** an. Das Notizbuch für das Ändern von Kurznamen wird geöffnet.
3. Wählen Sie auf der Seite 'Einstellungen' das Kontrollkästchen neben der jeweiligen Option aus, die Sie hinzufügen oder entfernen möchten. Sie können eine erforderliche Option nicht entfernen.
4. Um den Wert für eine Option anzugeben oder zu ändern, klicken Sie das Feld **Wert** für die entsprechende Option an. Abhängig von der Option können Sie entweder einen Wert aus der Liste auswählen, mehrere Werte durch Anklicken auswählen oder einen neuen Wert eingeben.
5. Klicken Sie zum Ändern des Kurznamens und Schließen des Notizbuchs **OK** an.

Um diese Task über eine Eingabeaufforderung der Befehlszeile auszuführen, verwenden Sie die Anweisung ALTER NICKNAME. Beispiel:

```
ALTER NICKNAME kurzname  
    OPTIONS (SET optionsname 'wert_der_optionszeichenfolge')
```

Der Kurzname DRUGDATA1 wird beispielsweise für die Datei mit Tabellenstruktur drugdata1.txt erstellt. Der ursprünglich in der Anweisung CREATE NICKNAME definierte vollständig qualifizierte Pfad lautete /user/pat/drugdata1.txt.

Um die Kurznamenoption FILE\_PATH zu ändern, geben Sie folgende Anweisung aus:

```
ALTER NICKNAME DRUGDATA1 OPTIONS (SET FILE_PATH '/usr/kelly/data/drugdata1.txt')
```

#### Zugehörige Tasks:

- „Ändern eines Kurznamens“ auf Seite 587

#### Zugehörige Referenzen:

- „Einschränkungen für das Ändern von Kurznamen“ auf Seite 589
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- Anhang F, „Kurznamenoptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 661

---

## Ändern der Spaltenoptionen von Kurznamen

Mit Hilfe von Parametern, den so genannten *Kurznamenspaltenoptionen*, geben Sie in den Anweisungen CREATE NICKNAME und ALTER NICKNAME Spalteninformationen an. Sie können alle diese Werte in Klein- oder in Großschreibung angeben.

Mit der Anweisung ALTER NICKNAME können Sie Kurznamenspaltenoptionen hinzufügen, definieren oder löschen.

#### Voraussetzungen:

Die Berechtigungs-ID, mit der die Anweisung eingegeben wird, muss über mindestens eines der folgenden Zugriffsrechte verfügen:

- Berechtigung SYSADM oder DBADM
- Zugriffsrecht ALTER für den Kurznamen, der in der Anweisung angegeben wurde
- Zugriffsrecht CONTROL für den Kurznamen, der in der Anweisung angegeben wurde
- Zugriffsrecht ALTERIN für das Schema, wenn der Schemaname des Kurznamens vorhanden ist
- Definierender Benutzer des Kurznamens, der in der Spalte DEFINER der Katalogsicht für den Kurznamen aufgezeichnet wurde

#### Einschränkungen:

Informationen zu den Einschränkungen beim Ändern von Kurznamen finden Sie im entsprechenden Abschnitt.

### Vorgehensweise:

Sie können Spaltennamen von der DB2-Steuerzentrale oder der DB2-Befehlszeile aus ändern.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Wählen Sie den Ordner **Kurznamen** aus.
2. Klicken Sie mit der rechten Maustaste auf den Kurznamen, den Sie ändern möchten, und klicken Sie auf **Ändern**. Das Notizbuch **Kurznamen ändern** wird geöffnet.
3. Wählen Sie auf der Seite **Kurznamen** die Spalte aus, die Sie ändern möchten, und klicken Sie auf **Ändern**. Das Fenster **Spalte ändern** wird geöffnet.
4. Wählen Sie die Spaltenoption aus, die Sie hinzufügen oder entfernen möchten.
5. Geben Sie für Optionen, die Sie hinzufügen oder ändern, den entsprechenden Optionswert an.
6. Klicken Sie auf **OK**, um die Spaltenoption zu ändern, und schließen Sie das Fenster.
7. Klicken Sie auf **OK**, um den Kurznamen zu ändern, und schließen Sie das Notizbuch.

Verwenden Sie die Anweisung `ALTER NICKNAME`, um diese Task von der Eingabeaufforderung der Befehlszeile aus auszuführen.

### Beispiel 1: Angeben der Spaltenoption `NUMERIC_STRING` bei relationalen Datenquellen:

Die Spaltenoption `NUMERIC_STRING` gilt für Zeichentypspalten (`CHAR` und `VARCHAR`). Angenommen, eine Datenquelle hat eine andere Sortierfolge als die zusammengeschlossene Datenbank. Normalerweise sortiert der Server mit zusammengeschlossenen Datenbanken in diesem Fall in der Datenquelle keine Spalten, die Zeichendaten enthalten. Stattdessen gibt er die Daten an die zusammengeschlossene Datenbank zurück und führt die Sortierung lokal aus. Nehmen Sie jedoch an, die Spalte ist ein Zeichendatentyp und enthält ausschließlich numerische Zeichen ('0', '1', ..., '9'). Dies können Sie durch die Zuordnung des Wertes 'Y' zu der Spaltenoption `NUMERIC_STRING` angeben. Dadurch kann das DB2 UDB-Abfrageoptimierungsprogramm die Sortierung in der Datenquelle durchführen. Wird die Sortierung fern durchgeführt, kann der Systemaufwand vermieden werden, der durch Sortieren der Daten auf dem Server mit zusammengeschlossenen Datenbanken entstehen würde.

Der Kurzname `ORA_INDSALES` gilt für eine Oracle-Tabelle mit dem Namen `INDONESIA_SALES`. Die Tabelle enthält die Spalte `POSTAL_CODE` mit dem Datentyp `VARCHAR`. Ursprünglich enthielt diese Spalte lediglich numerische Zeichen, und die Spaltenoption `NUMERIC_STRING` war auf 'Y' gesetzt. Allerdings enthält die Spalte nun eine Mischung aus numerischen und nicht numerischen Zeichen. Verwenden Sie die folgende Anweisung, um die Spaltenoption `NUMERIC_STRING` in 'N' zu ändern:

```
ALTER NICKNAME ORA_INDSALES ALTER COLUMN POSTAL_CODE  
OPTIONS (SET NUMERIC_STRING 'N')
```

### Beispiel 2: Angeben der Spaltenoption `VARCHAR_NO_TRAILING_BLANKS` bei relationalen Datenquellen:

Die Spaltenoption `VARCHAR_NO_TRAILING_BLANKS` kann zur Kennzeichnung bestimmter Spalten verwendet werden, die keine folgenden Leerzeichen enthalten.



Der SQL-Compiler wird diesen Faktor in der Einstellung berücksichtigen, wenn er alle für Spalten ausgeführten Operationen (z. B. Vergleichsoperationen) überprüft.

Der Kurzname `ORA_INDSALES` gilt für eine Oracle-Tabelle mit dem Namen `INDONESIA_SALES`. Die Tabelle enthält die Spalte `NAME` mit dem Datentyp `VARCHAR`. Die Spalte `NAME` verfügt nicht über folgende Leerzeichen. Verwenden Sie die folgende Anweisung, um die Option `VARCHAR_NO_TRAILING_BLANKS` zum Kurznamen hinzuzufügen:

```
ALTER NICKNAME ORA_INDSALES ALTER COLUMN NAME
  OPTIONS (ADD VARCHAR_NO_TRAILING_BLANKS 'Y')
```

### Beispiel 3: Angeben der Spaltenoption XPATH bei nicht relationalen Datenquellen:

Der Kurzname `EMPLOYEE` gilt für eine XML-Datenquelle. Für die Spalte `fname` wurde eine Spaltenoption `XPATH` angegeben. Verwenden Sie die folgende Anweisung, um in der Spaltenoption `XPATH` einen anderen Pfad zu definieren:

```
ALTER NICKNAME EMPLOYEE ALTER COLUMN fname
  OPTIONS (SET XPATH './@first')
```

#### Zugehörige Tasks:

- „Ändern eines Kurznamens“ auf Seite 587

#### Zugehörige Referenzen:

- „Einschränkungen für das Ändern von Kurznamen“ auf Seite 589
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- Anhang G, „Kurznamenspaltenoptionen für Systeme zusammengesetzter Datenbanken“, auf Seite 671

---

## Ändern eines lokalen Typs für ein Datenquellenobjekt

Wenn Sie einen Kurznamen erstellen, werden die Datentypen, die dem Datenquellenobjekt zugeordnet sind, in der zusammengesetzten Datenbank gespeichert. Für einige Datenquellen gibt der Wrapper die Datentypen an. Für andere Datenquellen müssen Sie bei der Erstellung des Kurznamens die Datentypen angeben.

Sie können für eine Spalte eines bestimmten Datenquellenobjekts einen lokalen Typ angeben. An Stelle der Anweisung `CREATE TYPE MAPPING` verwenden Sie die Anweisung `ALTER NICKNAME`.

**Achtung:** Beim Ändern des lokalen Datentyps kann es zu Fehlern oder Datenverlust kommen, wenn Sie den lokalen Datentyp für eine Spalte in einen Datentyp ändern, der sich erheblich vom fernen Datentyp unterscheidet.

#### Voraussetzungen:

Die Berechtigungs-ID, mit der die Anweisung eingegeben wird, muss über mindestens eines der folgenden Zugriffsrechte verfügen:

- Berechtigung `SYSADM` oder `DBADM`
- Zugriffsrecht `ALTER` für den Kurznamen, der in der Anweisung angegeben wurde
- Zugriffsrecht `CONTROL` für den Kurznamen, der in der Anweisung angegeben wurde



- Zugriffsrecht ALTERIN für das Schema, wenn der Schemaname des Kurznamens vorhanden ist

Die der Anweisung zugeordnete Berechtigungs-ID muss dem definierenden Benutzer des Kurznamens entsprechen, der in der Spalte DEFINER der Katalogsicht für den Kurznamen aufgezeichnet wurde.

### Einschränkungen:

Informationen zu den Einschränkungen beim Ändern von Kurznamen finden Sie im entsprechenden Abschnitt.

### Vorgehensweise:

Sie können den Datentyp von der DB2-Steuerzentrale oder der DB2-Befehlszeile aus ändern.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Wählen Sie den Ordner **Kurznamen** aus.
2. Klicken Sie mit der rechten Maustaste auf den Kurznamen, den Sie ändern möchten, und klicken Sie auf **Ändern**. Das Notizbuch **Kurznamen ändern** wird geöffnet.
3. Wählen Sie auf der Seite **Kurznamen** die Spalte aus, die Sie ändern möchten, und klicken Sie auf **Ändern**. Das Fenster **Spalte ändern** wird geöffnet.
4. Wählen Sie den Datentyp aus.
5. Klicken Sie auf **OK**, um den Datentyp zu ändern, und schließen Sie das Fenster.
6. Klicken Sie auf **OK**, um den Kurznamen zu ändern, und schließen Sie das Notizbuch.

Verwenden Sie die Anweisung ALTER NICKNAME, um diese Task von der Eingabeaufforderung der Befehlszeile aus auszuführen. Beispiel:

```
ALTER NICKNAME nickname ALTER COLUMN column_name
LOCAL TYPE data_type
```

Verwenden Sie die Klausel FOR BIT DATA in der Anweisung ALTER NICKNAME, um den Inhalt einer lokalen Spalte mit einem Zeichendatentyp wie Bit-Daten (binär) zu verarbeiten. Wenn Sie diese Klausel verwenden, um den lokalen Datentyp einer Spalte zu ändern, werden beim Austausch von Daten mit anderen Systemen keine Codepagekonvertierungen ausgeführt. Vergleiche werden binär und ohne Rücksicht auf die Sortierfolge der fernen Datenbank durchgeführt.

### Zugehörige Tasks:

- „Ändern eines Kurznamens“ auf Seite 587

### Zugehörige Referenzen:

- „Einschränkungen für das Ändern von Kurznamen“ auf Seite 589
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- Anhang H, „Standardmäßige progressive Datentypzuordnungen“, auf Seite 679
- „Ändern eines lokalen Typs für ein Datenquellenobjekt - Beispiele“ auf Seite 597
- „Unterstützte Datentypen für nicht relationale Datenquellen“ auf Seite 581

---

## Ändern eines lokalen Typs für ein Datenquellenobjekt - Beispiele

In den folgenden Beispielen sehen Sie, wie Änderungen an den Datentypen für ein Datenquellenobjekt vorgenommen werden.

### Beispiel: Eine numerische Datentypzuordnung:

In einer Oracle-Tabelle für Mitarbeiterinformationen wird die Spalte BONUS mit dem Datentyp NUMBER(32,3) definiert. Der Oracle-Datentyp NUMBER(32,3) wird standardmäßig dem DB2-Datentyp DOUBLE zugeordnet, einem Gleitkommazahl-datentyp mit doppelter Genauigkeit. Eine Abfrage, in der die Spalte BONUS enthalten ist, kann Werte zurückgeben, die folgendermaßen aussehen:

```
5.00000000000000E+002  
1.00000000000000E+003
```

Die Exponentialschreibweise gibt die Anzahl der Dezimalstellen und die Richtung an, in die das Dezimalzeichen verschoben werden soll. In diesem Beispiel gibt +002 an, dass das Dezimalzeichen zwei Stellen nach rechts bewegt werden soll, und +003, dass das Dezimalzeichen drei Stellen nach rechts bewegt werden soll.

Abfragen, in denen die Spalte BONUS enthalten ist, können Werte zurückgeben, die wie Dollarbeträge aussehen. Sie ändern die lokale Definition für die Spalte BONUS in der Tabelle vom Datentyp DOUBLE in den Datentyp DECIMAL. Verwenden Sie eine Genauigkeit und einen Maßstab, die das Format tatsächlicher Bonuszahlungen wiedergeben. Wenn beispielsweise der Dollaranteil der Bonuszahlungen nicht mehr als sechs Stellen umfasst, ordnen Sie NUMBER(32,3) zu DECIMAL(8,2) zu. Unter Vorgabe dieses neuen lokalen Typs geben Abfragen, die die Spalte BONUS enthalten, Werte zurück, die wie folgt aussehen:

```
500.00  
1000.00
```

Der Kurzname für die Oracle-Tabelle lautet ORASALES. Wenn Sie die Spalte BONUS in der Tabelle ORASALES dem DB2-Datentyp DECIMAL (8,2) zuordnen möchten, geben Sie die folgende Anweisung ALTER NICKNAME ein:

```
ALTER NICKNAME ORASALES ALTER COLUMN BONUS  
LOCAL TYPE DECIMAL(8,2)
```

**ORASALES**

Der Kurzname, den Sie für die Oracle-Tabelle definiert haben.

**ALTER COLUMN BONUS**

Der Name der Spalte, die in der Katalogsicht SYSCAT.COLUMNS der zusammenschlossenen Datenbank lokal definiert ist.

**LOCAL TYPE DECIMAL(8,2)**

Gibt den neuen lokalen Typ für die Spalte an.

| Diese Zuordnung wird nur auf die Spalte BONUS in der mit dem Kurznamen  
| ORASALES angegebenen Oracle-Tabelle angewendet. Alle anderen Oracle-Daten-  
| quellenobjekte, die die Spalte BONUS enthalten, verwenden die Standarddatentyp-  
| zuordnung für den Oracle-Datentyp NUMBER.

### Beispiel: Eine Datentypzuordnung:

Der Kurzname für eine Oracle-Tabelle mit dem Namen SALES lautet ORASALES. Die Tabelle SALES enthält eine Spalte mit dem Oracle-Datentyp DATE. Der Oracle-Datentyp DATE wird standardmäßig dem DB2-Datentyp TIMESTAMP zugeordnet. Allerdings möchten Sie vielleicht nur den Datumswert anzeigen, wenn Sie Daten aus dieser Spalte abrufen. Sie können dann den Kurznamen für die Tabelle SALES ändern, um den lokalen Typ in den DB2-Datentyp DATE zu ändern.

```
ALTER NICKNAME ORASALES ALTER COLUMN ORDER_DATE  
LOCAL TYPE DATE
```

### Beispiel: Eine Datentypzuordnung für eine nicht relationale Datenquelle:

Der Kurzname für eine Datei mit Tabellenstruktur mit dem Namen drugdata1.txt lautet DRUGDATA1. Die Datei drugdata1.txt enthält eine Spalte mit Namen pharmazeutischer Produkte. Der Spaltenname lautet DRUG. Die Spalte DRUG wurde ursprünglich mit CHAR(20) definiert. Die Spaltenlänge muss in CHAR(30) geändert werden. Sie können dann den Kurznamen der Datei drugdata1.txt ändern, um die Zuordnung zu ändern, dass die Länge korrekt ist:

```
ALTER NICKNAME DRUGDATA1 ALTER COLUMN DRUG  
LOCAL TYPE CHAR(30)
```

### Zugehörige Tasks:

- „Erstellen von Datentypzuordnungen“ in *Systeme zusammengeschlossener Datenbanken*
- „Ändern eines lokalen Typs für ein Datenquellenobjekt“ auf Seite 595

### Zugehörige Referenzen:

- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- „Einschränkungen für das Ändern von Kurznamen“ auf Seite 589

---

## Kapitel 29. DDL-Befehlsreferenz

Dieses Kapitel enthält ausführliche Erläuterungen zu den Syntaxanweisungen, -argumenten und -optionen für die in diesem Handbuch erörterten Wrapper-DDL-Befehle. Die Anweisungen sind nach Wrappern geordnet.

---

### BioRS-DDL-Referenzinformationen

#### Anweisungsoptionen für CREATE SERVER - BioRS-Wrapper

Folgende Optionen sind der Anweisung CREATE SERVER für BioRS zugeordnet:

**TYPE** Gibt den Servertyp an. Der Standardwert ist `BioRS`. Für den BioRS-Wrapper wird ausschließlich dieser Standardwert unterstützt. Diese Option braucht nicht angegeben zu werden.

**VERSION**

Gibt die Serverversion an. Der Standardwert ist `1.0`. Für den BioRS-Wrapper wird ausschließlich dieser Standardwert unterstützt. Diese Option braucht nicht angegeben zu werden.

**NODE**

Gibt den Hostnamen des Systems an, auf dem das BioRS-Abfragetool zur Verfügung steht. Der Standardwert ist `localhost` (lokaler Host).

**PORT** Gibt die Nummer des Ports an, der verwendet werden soll, um eine Verbindung zum BioRS-Server herzustellen. Der Standardwert ist `5014`.

**TIMEOUT**

Gibt die Zeit (in Minuten) an, die der BioRS-Wrapper auf Antworten vom BioRS-Server warten soll. Der Standardwert ist `10`. Diese Option muss angegeben werden.

**CASE\_SENSITIVE**

Gibt an, ob der BioRS-Server bei Namen die Groß-/Kleinschreibung berücksichtigen soll. Gültige Werte sind 'Y' (ja) und 'N' (nein). Der Standardwert ist 'Y'.

Bei BioRS wird die Groß-/Kleinschreibung der auf der BioRS-Servermaschine gespeicherten Daten durch einen Konfigurationsparameter gesteuert. Die Option `CASE_SENSITIVE` ist das von DB2 Information Integrator verwendete Gegenstück zum BioRS-Systemkonfigurationsparameter. Die Konfigurationseinstellungen für die Groß-/Kleinschreibung auf dem BioRS-Server müssen im BioRS-System und in DB2 Information Integrator synchronisiert werden. Werden die Konfigurationseinstellungen für die Groß-/Kleinschreibung zwischen BioRS und DB2 Information Integrator nicht synchronisiert, führt dies zu Fehlern, wenn versucht wird, über DB2 Information Integrator auf BioRS-Daten zuzugreifen.

**Wichtig:** Nach Erstellung eines neuen BioRS-Servers unter DB2 Information Integrator kann die Option `CASE_SENSITIVE` nicht mehr geändert oder gelöscht werden. Ist eine Änderung der Option `CASE_SENSITIVE` erforderlich, müssen Sie den gesamten Server zunächst löschen und anschließend erneut erstellen. Wenn Sie den BioRS-Server löschen, müssen

Sie auch alle zugehörigen BioRS-Kurznamen erneut erstellen. DB2 Information Integrator löscht automatisch alle Kurznamen, die einem gelöschten Server zugeordnet sind.

**Zugehörige Tasks:**

- „Registrieren der Serverdefinition für eine BioRS-Datenquelle“ auf Seite 83
- „Registrieren von Kurznamen für BioRS-Datenquellen“ auf Seite 84

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- „Anweisungssyntax für CREATE NICKNAME - BioRS-Wrapper“ auf Seite 601

## Anweisungsoptionen für CREATE USER MAPPING - BioRS-Wrapper

**GUEST**

Gibt an, ob Operationen unter dem BioRS-Authentifizierungsmechanismus für Gäste (GUEST) auf dem BioRS-Server ausgeführt werden sollen. Gültige Werte sind 'Y' (ja) und 'N' (nein). Der Standardwert ist 'Y'.

Wird diese Option auf 'Y' gesetzt, wird für den Zugriff auf den BioRS-Server die Gastauthentifizierung für den betreffenden Benutzer von DB2 Information Integrator verwendet.

Wird diese Option auf 'N' gesetzt, muss der betreffende Benutzer von DB2 Information Integrator eine BioRS-Berechtigungs-ID und ein BioRS-Kennwort angeben, um auf den BioRS-Server zugreifen zu können.

Wird keine Benutzerzuordnung erstellt oder wird eine Benutzerzuordnung ohne Angabe von Optionen erstellt, wird für den Zugriff auf den BioRS-Server die Gastauthentifizierung für den betreffenden Benutzer von DB2 Information Integrator verwendet.

**REMOTE\_AUTHID**

Gibt eine Benutzer-ID an, die den betreffenden DB2-Benutzer für den Zugriff auf BioRS-Datenquellen berechtigt. Diese ferne ID muss über das Format verfügen, das von der BioRS-Anwendung erwartet wird. Diese Option ist erforderlich, wenn die Option GUEST auf 'N' gesetzt ist.

**REMOTE\_PASSWORD**

Gibt das Kennwort für diese ferne ID an. Diese Option ist erforderlich, wenn die Option GUEST auf 'N' gesetzt ist.

**Beispiel:**

Die folgende Anweisung CREATE USER MAPPING ordnet den Benutzer Charlie dem Benutzer Charlene auf dem Server Biors\_Server1 zu.

```
CREATE USER MAPPING FOR Charlie SERVER Biors_Server1
  OPTIONS(GUEST 'N', REMOTE_AUTHID 'Charlene', REMOTE_PASSWORD 'Charlene_pw');
```

**Zugehörige Tasks:**

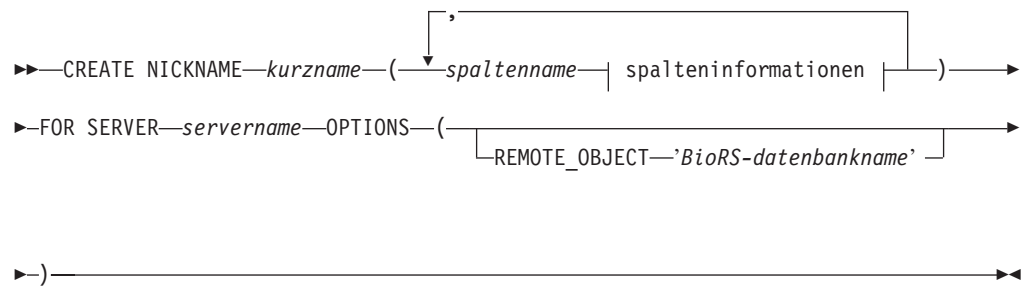
- „Registrieren von Benutzerzuordnungen für BioRS-Datenquellen“ auf Seite 83

**Zugehörige Referenzen:**

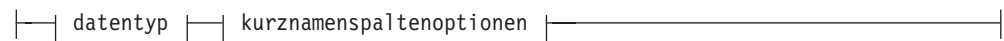
- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

## Anweisungssyntax für CREATE NICKNAME - BioRS-Wrapper

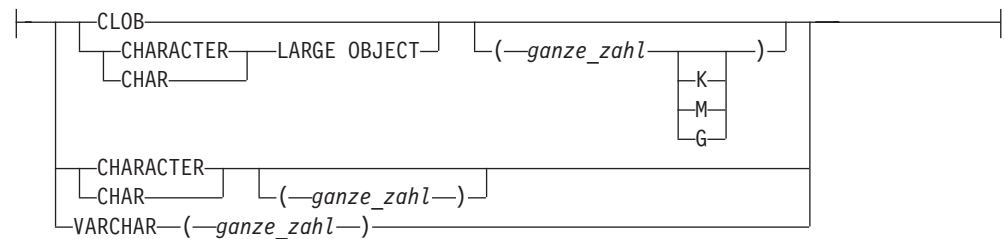
Die Syntax für die Anweisung CREATE NICKNAME sieht wie folgt aus:



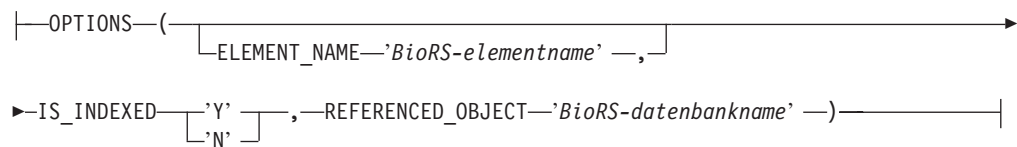
### spalteninformationen:



### datentyp:



### kurznamenspaltenoptionen:



### Kurznamenspaltenoptionen

Werte für Kurznamenspaltenoptionen müssen in einfache Anführungszeichen eingeschlossen werden.

#### ELEMENT\_NAME

Gibt den Namen des BioRS-Elements an. Die Groß-/Kleinschreibung dieses Namens hängt von der Groß-/Kleinschreibung des BioRS-Servers und vom Wert der Serveroption CASE\_SENSITIVE ab. Der Name des BioRS-Elements braucht nur dann angegeben zu werden, wenn er sich vom Namen der Spalte unterscheidet.

#### IS\_INDEXED

Gibt an, ob die entsprechende Spalte indiziert ist, d. h. ob auf die Spalte in einem Vergleichselement verwiesen werden kann. Gültige Werte sind 'Y'

(ja) und 'N' (nein). Der Wert 'Y' kann nur für Spalten angegeben werden, deren entsprechendes Element vom BioRS-Server indexiert ist.

Bei der Erstellung eines Kurznamens wird diese Option automatisch mit dem Wert 'Y' allen Spalten hinzugefügt, die einem indexierten BioRS-Element entsprechen.

#### **REFERENCED\_OBJECT**

Diese Option ist nur für Spalten gültig, deren BioRS-Datentyp 'Reference' (Verweis) lautet. Diese Option gibt den Namen der BioRS-Datenbank an, auf die die aktuelle Spalte verweist. Die Groß-/Kleinschreibung dieses Namens hängt von der Groß-/Kleinschreibung des BioRS-Servers und vom Wert der Serveroption CASE\_SENSITIVE ab.

#### **Kurznamenoptionen**

Optionswerte für Kurznamen müssen in einfache Anführungszeichen eingeschlossen werden.

#### **REMOTE\_OBJECT**

Gibt den Namen der BioRS-Datenbank an, die dem Kurznamen zugeordnet ist. Dieser Name legt das Schema und die BioRS-Datenbank für den Kurznamen fest. Außerdem gibt dieser Name die Abhängigkeit zwischen dem Kurznamen und anderen Kurznamen an. Die Groß-/Kleinschreibung dieses Namens hängt von der Groß-/Kleinschreibung des BioRS-Servers und vom Wert der Serveroption CASE\_SENSITIVE ab.

**Wichtig:** Dieser Name kann nicht mit der Anweisung ALTER NICKNAME geändert oder gelöscht werden. Ändert sich der Name der BioRS-Datenbank, der in dieser Option verwendet wird, müssen Sie den gesamten Kurznamen zunächst löschen und dann erneut erstellen.

#### **Zugehörige Tasks:**

- „Registrieren von Kurznamen für BioRS-Datenquellen“ auf Seite 84

#### **Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den BioRS-Wrapper“ auf Seite 86

---

## **BLAST-DDL-Referenzinformationen**

### **CREATE SERVER-Anweisungsargumente - BLAST-Wrapper**

#### **CREATE SERVER-Argumente für den BLAST-Wrapper:**

**TYPE** Legt den Typ der BLAST-Suche fest, die mit Hilfe des angegebenen Servers ausgeführt wird. Dieses Argument ist erforderlich. Es muss auf einen der folgenden Werte eingestellt werden: blastn, blastp, blastx, tblastn, tblastx.

#### **VERSION**

Gibt die Version des verwendeten Servers an. Sie sollte der Version der ausgeführten Datei 'blastall' entsprechen. Dieses Argument ist erforderlich.

#### **WRAPPER**

Gibt den Namen des Wrappers an, die Sie mit der Anweisung CREATE WRAPPER registriert haben. Dieses Argument ist erforderlich.

## Serveroptionen für den BLAST-Wrapper:

Folgende Optionen können für BLAST in der Anweisung CREATE SERVER angegeben werden:

- DAEMON\_PORT
- NODE
- USE\_CLOB\_SEQUENCE

## Zugehörige Tasks:

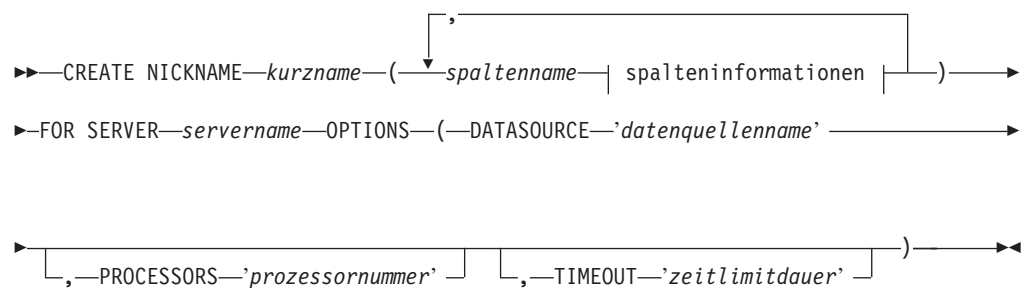
- „Registrieren des Servers für BLAST-Datenquellen“ auf Seite 122

## Zugehörige Referenzen:

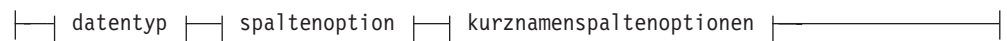
- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641

## CREATE NICKNAME-Anweisungssyntax - BLAST-Wrapper

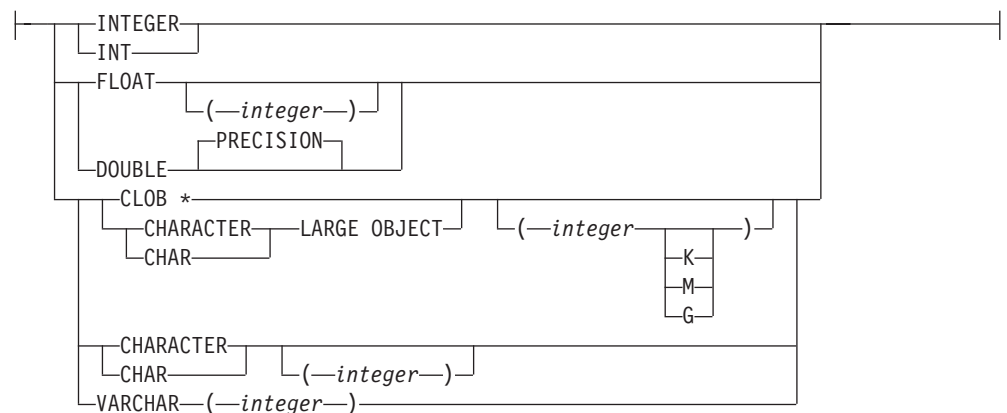
Die Syntax für die Anweisung CREATE NICKNAME sieht wie folgt aus:



## spalteninformationen:



## datentyp:





### spaltenoption:

|-----|  
| NOT NULL |

### kurznamenoptionen:

|----->  
OPTIONS (INDEX 'indexnummer' , DELIMITER 'begrenzer'
DEFAULT 'neuer\_standardwert'

**Einschränkung:** Beim BLAST-Wrapper ist die Länge eines CLOBs auf 5 Megabyte (5 MB) beschränkt.

Für BLAST-Datenquellen können 2 Typen von Optionen in der Anweisung CREATE NICKNAME angegeben werden:

- Optionen für Kurznamen
- Optionen für Kurznamen

### Zugehörige Tasks:

- „Registrieren von Kurznamen für BLAST-Datenquellen“ auf Seite 123

### Zugehörige Referenzen:

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- Anhang G, „Kurznamenoptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 671
- Anhang F, „Kurznamenoptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 661
- „Anweisung CREATE NICKNAME - Beispiele für den BLAST-Wrapper“ auf Seite 130

---

## Documentum-DDL-Referenzinformationen

### CREATE SERVER-Anweisungsargumente und -optionen - Documentum-Wrapper

Folgende Argumente sind der Anweisung CREATE SERVER für Documentum zugeordnet:

**TYPE** Gibt den Typ der Datenquelle an. Für Documentum lautet der Typ DCTM. Dieses Argument ist erforderlich.

#### VERSION

Gibt die Version der Datenquelle an. Für EDMS98 lautet der Wert '3'. Für 4i lautet der Wert '4'. Dieses Argument ist erforderlich.

#### WRAPPER

Gibt den Namen des Wrappers an, der diesem Server zugeordnet ist. Dieses Argument ist erforderlich.

Folgende Optionen sind der Anweisung CREATE SERVER für Documentum zugeordnet:

#### **CONTENT\_DIR**

Gibt den Namen des Stammverzeichnisses (Root) an, auf das lokaler Zugriff besteht, um die durch die Pseudospalten GET\_FILE, GET\_FILE\_DEL, GET\_RENDITION und GET\_RENDITION\_DEL abgerufenen Inhaltsdateien zu speichern. Auf diese Verzeichnis muss Schreibzugriff für alle Benutzer bestehen, die diese Pseudospalten verwenden können. Der Standardwert lautet /tmp. Diese Option ist wahlfrei.

#### **NODE**

Gibt den tatsächlichen Namen der Documentum-Doabase an. Diese Option ist erforderlich.

#### **OS\_TYPE**

Gibt das Betriebssystem des Doabase-Servers an. Gültige Werte sind AIX, SOLARIS und WINDOWS. Diese Option ist erforderlich.

#### **RDBMS\_TYPE**

Gibt das Verwaltungssystem für relationale Datenbanken (RDBMS) an, das von der Doabase verwendet wird. Gültige Werte sind DB2, INFORMIX, ORACLE, SQLSERVER oder SYBASE. Diese Option ist erforderlich.

#### **TRANSACTIONS**

Gibt den Servertransaktionsmodus an. Gültige Werte sind:

- NONE - es werden keine Transaktionen aktiviert.
- QUERY - Transaktionen werden nur für Dctm\_Query-Methoden aktiviert.
- ALL - Transaktionen werden für die Dctm\_Query-Methode aktiviert. ALL hat im vorliegenden Release dieselbe Funktion wie QUERY.

Der Standardwert lautet QUERY. Diese Option ist wahlfrei.

#### **Zugehörige Tasks:**

- „Registrieren des Servers für Documentum-Datenquellen“ auf Seite 198

#### **Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

## **CREATE USER MAPPING-Anweisungsoptionen - Documentum-Wrapper**

#### **Optionsdefinitionen:**

##### **REMOTE\_AUTHID**

Ihre Berechtigungskennung auf dem fernen Server.

##### **REMOTE\_PASSWORD**

Ihr Kennwort auf dem fernen Server.

##### **REMOTE\_DOMAIN**

Ihre Windows-Netzwerkdomäne auf dem fernen Server. Gültig nur für Windows-Plattformen.

#### **Zugehörige Tasks:**

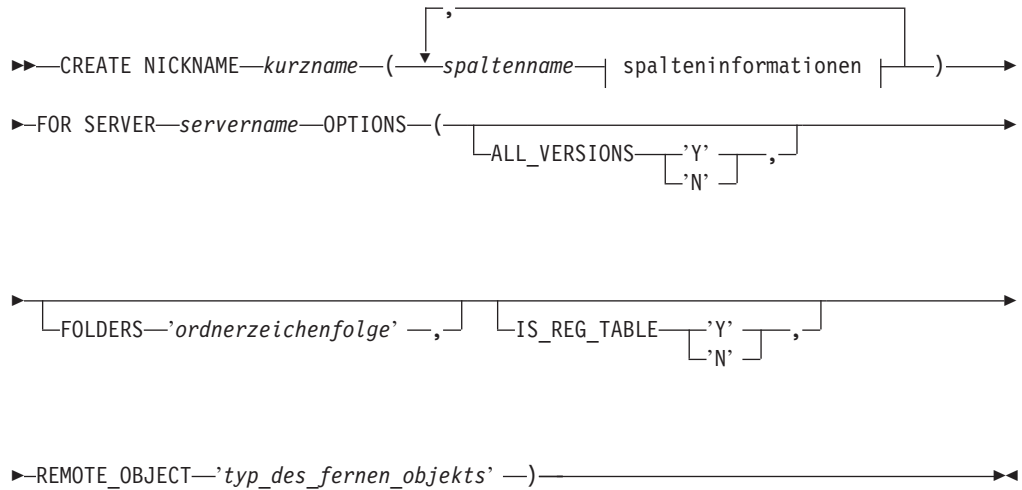
- „Registrieren der Benutzerzuordnungen für Documentum-Datenquellen“ auf Seite 198

#### **Zugehörige Referenzen:**

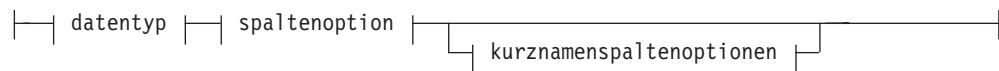
- „CREATE USER MAPPING statement“ in *SQL Reference, Volume 2*

# CREATE NICKNAME-Anweisungssyntax - Documentum-Wrapper

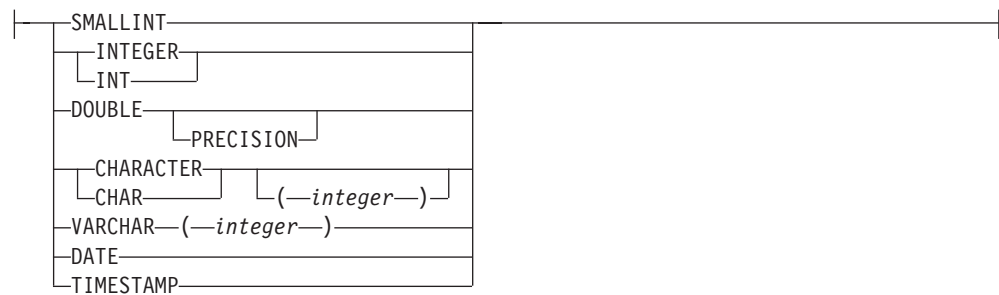
Die Syntax für die Anweisung CREATE NICKNAME für Documentum lautet:



## spalteninformationen:



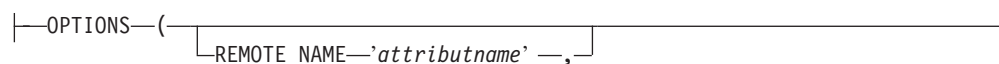
## datentyp:

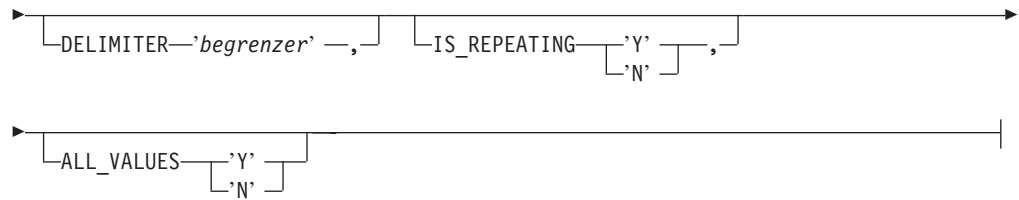


## spaltenoption:



## kurznamenspaltenoptionen:





Folgende Spaltenoptionen sind der Anweisung CREATE NICKNAME für Documentum zugeordnet:

#### NOT NULL

Alle Spalten mit nur einem Wert, mit Ausnahme der als TIMESTAMP und DATE definierten, müssen als NOT NULL definiert werden. Wiederholungsattribute dürfen in Kurznamen nicht als NOT NULL definiert werden.

Folgende Kurznamenoptionen sind der Anweisung CREATE NICKNAME für Documentum zugeordnet:

Werte für Kurznamenoptionen müssen in einfache Anführungszeichen eingeschlossen werden.

#### ALL\_VALUES

Gibt an, dass alle Werte eines Wiederholungsattributs durch den angegebenen Begrenzer getrennt zurückgegeben werden. Fehlt diese Option oder hat sie den Wert 'N', wird nur der letzte Wert eines Wiederholungsattributs zurückgegeben. Wie unter DELIMITER erwähnt, kann ALL\_VALUES nur für VARCHAR-Spalten angegeben werden, bei denen für die Option IS\_REPEATING der Wert 'Y' angegeben ist. Wurde für diese Spalten IS\_REG\_TABLE = 'Y' angegeben, ist die Angabe von ALL\_VALUES ungültig.

#### DELIMITER

Gibt die Begrenzerzeichenfolge an, die verwendet werden muss, wenn mehrere Werte eines Wiederholungsattributs verknüpft werden sollen. Der Begrenzer kann aus einem oder mehreren Zeichen bestehen. Das Standardzeichen für den Begrenzer ist ein Komma. Diese Option ist nur für Attribute von Objekten mit dem Datentyp VARCHAR gültig, für die die Option IS\_REPEATING auf den Wert 'Y' gesetzt ist. Diese Option ist wahlfrei.

#### IS\_REPEATING

Gibt an, ob für die Spalte mehrere Werte zulässig sind. Gültige Werte sind 'Y' und 'N'. Der Standardwert ist 'N'. Diese Option ist wahlfrei.

Für diese Spalten wird nur der letzte Wert zurückgegeben:

- Nicht-VARCHAR-Wiederholungsattribute
- VARCHAR-Spalten bei Angabe von ALL\_VALUES 'N'

Um diese Einschränkung zu umgehen, können Sie eine doppelte Definition für die Spalte des Wiederholungsattributs erstellen.

#### REMOTE\_NAME

Gibt den Namen des entsprechenden Documentum-Attributs bzw. der entsprechenden Documentum-Spalte an. Mit dieser Option werden Namen von fernen Attributen oder Spalten lokalen DB2-Spaltennamen zugeordnet. Standardmäßig wird der DB2-Spaltenname verwendet. Diese Option ist wahlfrei.

Folgende Kurznamenoptionen sind der Anweisung CREATE NICKNAME für Documentum zugeordnet:

Optionswerte für Kurznamen müssen in einfache Anführungszeichen eingeschlossen werden.

#### **ALL\_VERSIONS**

Gibt an, ob alle Objektversionen durchsucht werden. Gültige Werte sind 'y', 'Y', 'n' und 'N'. Der Standardwert 'N' bedeutet, dass nur die aktuellen Objektversionen Teil der Abfrageverarbeitung sind. Diese Option ist nicht gültig, wenn IS\_REG\_TABLE = 'Y' definiert wird. Diese Option ist wahlfrei.

#### **FOLDERS**

Gibt eine Zeichenfolge an, die ein oder mehrere logisch kombinierte und syntaktisch korrekte Documentum-FOLDER-Vergleichselemente enthält. Durch die Angabe von FOLDER-Vergleichselementen wird die Gruppe von Dokumenten, die durch diesen Kurznamen repräsentiert werden, auf die Dokumenten in den angegebenen Ordnern begrenzt.

Wenn Sie diese Option angeben, müssen Sie den gesamten Wert der Option FOLDERS in einfache Anführungszeichen einschließen; verwenden Sie innerhalb der Zeichenfolge doppelte Anführungszeichen an Stelle der einfachen Anführungszeichen.

Beispiel: Für die folgende Angabe:

```
FOLDER('/Tools',DESCEND) OR FOLDER('/Cars')
```

muss die folgende Option FOLDERS angegeben werden:

```
FOLDERS 'FOLDER("/Tools",DESCEND) OR FOLDER("/Cars")'
```

Diese Option ist nicht gültig, wenn IS\_REG\_TABLE = 'Y' definiert wird. Diese Option ist wahlfrei.

#### **IS\_REG\_TABLE**

Gibt an, ob das in der Option REMOTE\_OBJECT angegebene Objekt eine registrierte Documentum-Tabelle ist. Gültige Werte sind 'y', 'Y', 'n' und 'N'. Der Standardwert ist 'N'. Diese Option ist wahlfrei.

Sie können einen Kurznamen nicht von einem Documentum-Objekt in eine registrierte Tabelle ändern (oder umgekehrt), indem Sie diese Option in der Anweisung ALTER NICKNAME ändern. Stattdessen müssen Sie den Kurznamen löschen (DROP) und erneut erstellen (CREATE).

#### **REMOTE\_OBJECT**

Gibt den Namen des Documentum-Objekttyps an, der dem Kurznamen zugeordnet ist. Der Name kann ein beliebiger Documentum-Objekttyp oder eine beliebige registrierte Documentum-Tabelle sein. Einer registrierten Tabelle muss der Name des Tabelleneigners voran gestellt werden. Ist der Docbase-Eigner auch Eigner der registrierten Tabelle, kann dm\_dbo als Eigernamen verwendet werden. Diese Option ist erforderlich.

Die Verwendung der Anweisung ALTER NICKNAME zum Ändern des Wertes der Option REMOTE\_OBJECT verursacht Fehler, wenn die Struktur des neuen Objekts nicht mit der Struktur des ursprünglichen Objekts übereinstimmt.

#### **Zugehörige Tasks:**

- „Registrieren von Kurznamen für Documentum-Datenquellen“ auf Seite 199

**Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiel für Documentum-Wrapper“ auf Seite 204

---

## Excel-DDL-Referenzinformationen

### CREATE SERVER-Anweisungsargumente - Excel-Wrapper

Folgende Argumente sind der Anweisung CREATE SERVER für Excel zugeordnet:

**WRAPPER**

Gibt den Namen des Wrappers an, der in der entsprechenden Anweisung CREATE WRAPPER registriert wurde. Dieses Argument ist erforderlich.

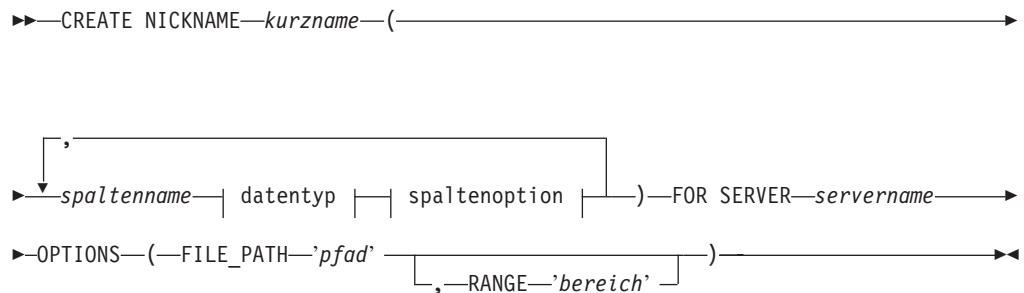
**Zugehörige Tasks:**

- „Registrieren des Servers für eine Excel-Datenquelle“ auf Seite 253

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*

### CREATE NICKNAME-Anweisungssyntax - Excel-Wrapper



**datentyp:**



**spaltenoption:**



Dabei gilt:

### FOR SERVER

Identifiziert den Server, der in der entsprechenden Anweisung CREATE SERVER registriert wurde. Dieser Server wird für den Zugriff auf die Excel-Tabelle verwendet. Geben Sie den Servernamen an.

Nachfolgend sind die Optionen der Anweisung CREATE NICKNAME für Excel beschrieben:

### FILE\_PATH

Gibt den vollständig qualifizierten Verzeichnispfad und Dateinamen für die Excel-Tabelle an, auf die zugegriffen werden soll.

Die Datentypen müssen innerhalb einer Spalte konsistent sein. Die Spalten- datentypen müssen während des Registrierungsprozesses für Kurznamen beschrieben werden.

Die Excel-Wrapper können lediglich auf die primäre Tabelle innerhalb eines Excel-Workbooks zugreifen.

Leere Zellen in den Tabellen werden als NULL interpretiert.

Maximal zehn aufeinanderfolgende leere Zeilen in der Tabelle können in den Datensatz aufgenommen werden. Mehr als zehn aufeinanderfolgende leere Zeilen werden als Ende des Datensatzes interpretiert.

Leere Spalten können in der Tabelle vorhanden sein. Diese Spalten müssen jedoch registriert und als gültige Felder beschrieben sein, selbst dann, wenn sie nicht verwendet werden.

Die Datenbank-Codepage muss mit dem Zeichensatz der Datei übereinstimmen. Anderenfalls kann es zu unerwarteten Ergebnissen kommen.

### RANGE

Gibt einen Bereich von zu verwendenden Zellen in der Datenquelle an. Diese Option ist nicht erforderlich.

Jeder Syntax- oder Semantikfehler im Wert der Bereichsoption führt zu einer Nachricht des Typs SQL1882E. Folgende Fehler können auftreten:

- Die Bezugswerte oben links und unten rechts sind nicht ordnungsgemäß ausgerichtet. Eine nicht ordnungsgemäße Ausrichtung bedeutet, dass der Bezugswert der Zelle oben links sich entweder unterhalb oder rechts des Bezugswertes der Zelle unten rechts befindet.
- Die Anzahl der vom Bereichswert zugeordneten Spalten stimmt nicht mit der Anzahl der in der Anweisung CREATE NICKNAME angegebenen Spalten überein.
- Ein ungültiges Zeichen oder sonstiger Syntaxfehler wurde gefunden.

Es folgt ein Beispiel für die Kurznamenoption RANGE:

```
CREATE NICKNAME excel2
(c1 VARCHAR (10),
c2 VARCHAR (10),
c3 VARCHAR (10),
c4 VARCHAR (10)
) FOR SERVER excel_server
OPTIONS (FILE_PATH 'C:\My Documents\test2.xls',
RANGE 'B2:E5');
```

In diesem Beispiel ist **B2** das Feld oben links in einem Zellbereich, und **E5** ist das Feld unten rechts in diesem Zellbereich. Der Buchstabe *B* in B2 ist die Spaltenbelegung. Die Zahl 2 in B2 ist die Zeilenzahl.

Die Belegung unten rechts im Bereich kann übergangen werden. In diesem Fall wird die gültige Zeile unten rechts verwendet. Wird der Wert oben links übergangen, lautet der Wert *A1*. Gibt der Bereich mehr Zeilen an, als tatsächlich in der Tabelle vorhanden sind, wird die tatsächliche Anzahl der Zeilen verwendet.

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Excel-Datenquellen“ auf Seite 253

**Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

## Extended Search-DDL-Referenzinformationen

### CREATE WRAPPER-Anweisungssyntax - Extended Search-Wrapper

►► CREATE WRAPPER *wrappername* LIBRARY *'bibliotheksname'* ◀◀

**WRAPPER**

Gibt einen eindeutigen Namen für diesen Extended Search-Wrapper an.

**LIBRARY**

Gibt einen der folgenden plattformabhängigen Bibliotheksnamen an:

- Windows: db2uies.dll
- AIX: libdb2uies.a

**Zugehörige Tasks:**

- „Registrieren des Extended Search-Wrappers“ auf Seite 271

**Zugehörige Referenzen:**

- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

### CREATE SERVER-Anweisungssyntax - Extended Search-Wrapper

►► CREATE SERVER *servername* WRAPPER *wrappername* OPTIONS ( ◀◀

► ES\_HOST *'hostname'* , ◀◀

◀◀ ES\_PORT *'portnummer'* , ◀◀

◀◀ ES\_TRACING [ OFF ] , ◀◀

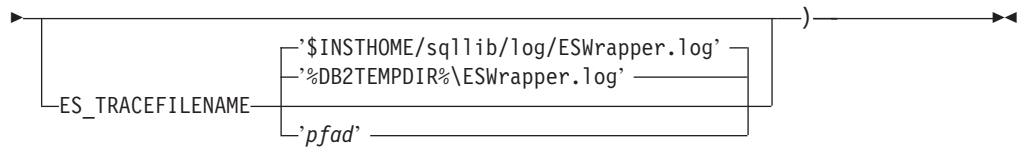
◀◀ ES\_TRACELEVEL [ C ] , ◀◀

◀◀ [ N ]

◀◀ [ W ]

◀◀ [ I ]





**SERVER**

Gibt einen eindeutigen Namen für diese Serverdefinition an. Dieser Parameter ist obligatorisch.

**WRAPPER**

Gibt den Namen eines zu einem früheren Zeitpunkt registrierten Extended Search- Wrappers an, den Sie mit dieser Serverdefinition verwenden möchten. Dieser Parameter ist obligatorisch.

**ES\_HOST**

Gibt einen vollständig qualifizierten Hostnamen oder eine IP-Adresse des zu durchsuchenden Extended Search-Servers an. Diese Option ist erforderlich.

**ES\_PORT**

Gibt die Portnummer an, unter der dieser Extended Search-Server Abfragen empfängt. Übergehen Sie diese Option, wird der Standardwert 6001 verwendet.

**ES\_TRACING**

Gibt an, ob die Tracefunktion für Fehlermeldungen, Warnungen und Informationsnachrichten aktiviert werden soll, die vom fernen Extended Search-Server generiert werden. Der Standardwert OFF bedeutet, dass keine Tracenachrichten protokolliert werden.

**ES\_TRACELEVEL**

Ist die Tracefunktion aktiviert, gibt diese Option die Typen von Nachrichten an, die in die Protokolldatei geschrieben werden. Bei Verwendung des Standardwertes C werden nur kritische Nachrichten protokolliert. Sie können die folgenden Ebenen der Tracefunktion voneinander unabhängig aktivieren und inaktivieren:

- C - Kritische Fehlernachrichten
- N - Nicht kritische Fehlernachrichten
- W - Warnungen
- I - Informationsnachrichten

Beispiel:

```
ES_TRACELEVEL 'W'
ES_TRACELEVEL 'CN'
ES_TRACELEVEL 'CNWI'
```

**ES\_TRACEFILENAME**

Ist die Tracefunktion aktiviert, gibt diese Option den Namen eines Verzeichnisses und einer Datei an, in die die Nachrichten geschrieben werden. Wenn Sie diese Option ausschließen, gelten folgende Standardwerte:

- Unter UNIX: \$INSTHOME/sql11ib/log/ESWrapper.log
- Unter Windows: %DB2TEMPDIR%\ESWrapper.log

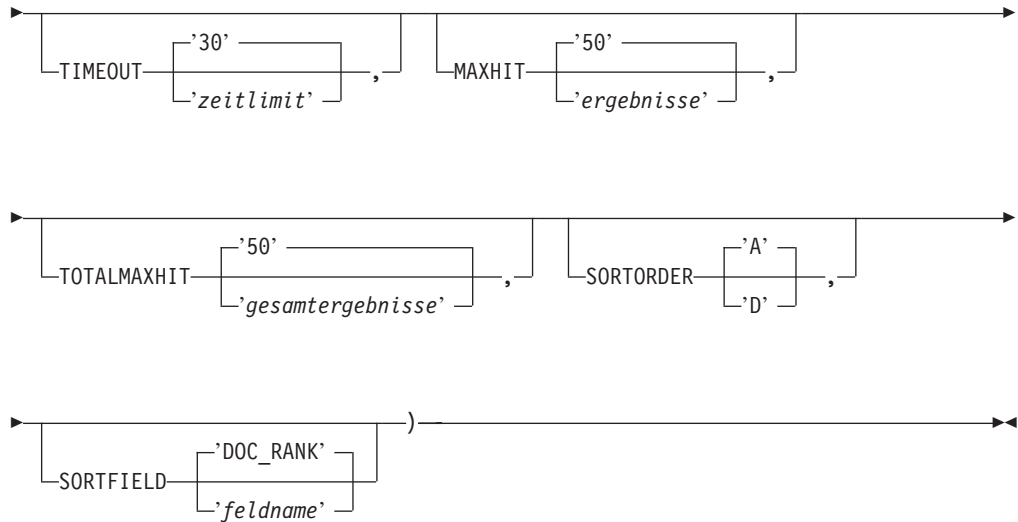
**Zugehörige Tasks:**

- „Registrieren des Servers für Extended Search-Datenquellen“ auf Seite 272

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*





**datentyp:**



**CREATE NICKNAME**

Gibt einen eindeutigen Namen für diese Extended Search-Kurznamentabelle an. Dieser Name muss sich von allen anderen Kurznamen in dem Schema unterscheiden, für das er definiert wird. Dieser Parameter ist obligatorisch.

*spaltenname*

Gibt einen oder mehrere benutzerdefinierte Spaltennamen an. Der Spaltenname muss mit dem Namen eines nativen oder zugeordneten Feldes übereinstimmen, das in der Extended Search-Konfigurationsdatenbank definiert wird. Dieser Parameter ist optional.

*datentyp*

Gibt den SQL-Datentyp der genannten Spalte an. Dieser Datentyp muss dem Datentyp entsprechen, der für dieses Feld in der Extended Search-Konfigurationsdatenbank definiert ist. Um beispielsweise ein Feld des Datentyps einer Zeichenfolge in einer Extended Search-Datenquelle zu suchen, definieren Sie eine Spalte VARCHAR für dieses Feld in der Kurznamentabelle. Wenn Sie einen *spaltennamen* angeben, ist dieser Parameter erforderlich.

**FOR SERVER**

Gibt den Namen einer zu einem früheren Zeitpunkt registrierten Serverdefinition an, die für den zu durchsuchenden Extended Search-Server erstellt wurde. Dieser Parameter ist obligatorisch.

**APPLICATIONID**

Gibt den Namen der zu durchsuchenden Extended Search-Anwendung an. Dieser Name muss in der Extended Search-Konfigurationsdatenbank vorhanden sein. Dieser Parameter ist obligatorisch.

**CATEGORY**

Gibt eine oder mehrere Extended Search-Kategorien an, die Sie durchsuchen möchten. Wenn Sie diese Option übergehen, müssen Sie zumindest einen Datenquellennamen angeben. Um mehrere Kategorien anzugeben, begrenzen Sie die Kategorienamen durch ein Semikolon. Beispiel:

```
CATEGORY 'LotusNotes;MSAccess;LDAP'
```

**DATASOURCE**

Gibt eine oder mehrere Extended Search-Datenquellen an, die Sie durchsuchen möchten. Wenn Sie diese Option übergehen, müssen Sie zumindest einen Kategorienamen angeben. Um mehrere Datenquellen anzugeben, begrenzen Sie die Datenquellennamen durch ein Semikolon. Beispiel:

```
DATASOURCE 'AltaVista;Google!;CNN'
```

**VERTICAL\_TABLE**

Gibt das Darstellungsformat für Suchergebnisse an. Bei Angabe von YES gibt Extended Search alle als rückgebbar konfigurierten Felder zurück, nicht nur die benutzerdefinierten Spalten. Der Wrapper speichert die Ergebnisse in der Kurznamentabelle als vertikale Liste mit Spaltennamen. Der Standardwert ist NO.

**TIMEOUT**

Ein Wert für eine ganze Zahl, der die Antwortzeit eines Servers in Sekunden angibt, bevor der Zeitlimit einer Abfrage überschritten wird. Diese Option ist wahlfrei. Der Standardwert ist 30.

**MAXHIT**

Ein Wert für eine ganze Zahl, der die maximale Anzahl von Ergebnissen angibt, die von jeder durchsuchten Quelle zurückgegeben werden kann. Diese Option ist wahlfrei. Der Standardwert ist 50.

**TOTALMAXHIT**

Ein Wert für eine ganze Zahl, der die maximale Anzahl von Ergebnissen angibt, die von allen durchsuchten Quellen zurückgegeben werden kann. Der Wrapper stellt diese Ergebnisse als eine einzige Ergebnismenge zusammen. Diese Option ist wahlfrei. Der Standardwert ist 50.

**SORTORDER**

Gibt eine Sortierfolge für die Rückgabe der Suchergebnisse an, entweder aufsteigend (A) oder absteigend (D). Der Standardwert ist A.

**SORTFIELD**

Gibt den Namen eines Feldes an, nach dem die Suchergebnisse sortiert werden sollen. Der Standardwert, DOC\_RANK, ist ein Feld, das Extended Search zur Gewichtung der Relevanz eines Ergebnisdokuments verwendet. Wenn Sie einen anderen Feldnamen angeben, stellen Sie sicher, dass der Name in den zu durchsuchenden Quellen vorhanden ist.

**Zugehörige Konzepte:**

- „Kurznamen für Extended Search“ auf Seite 266
- „Vertikale Tabellen für Extended Search“ auf Seite 267

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Extended Search-Datenquellen“ auf Seite 274

**Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Extended Search-Wrapper - Beispielabfragen“ auf Seite 278

## CREATE FUNCTION-Anweisungssyntax - Extended Search-Wrapper

Die Syntax für die Anweisung CREATE FUNCTION sieht wie folgt aus:

```

▶▶ CREATE FUNCTION ESWRAPPER.ES_SEARCH—(—INTEGER—,—VARCHAR(1024)—)—————▶
▶ RETURNS INTEGER AS TEMPLATE DETERMINISTIC NO EXTERNAL ACTION—————▶▶

```

**INTEGER**

Definiert den Verweisparameter der Abfrage. Dieser Parameter muss in einer Abfrage den Namen der Spalte INTEGER angeben, die in der Kurznamens-tabelle definiert ist, für die diese angepasste Funktion aufgerufen wird. Der Wert muss eine Binde-spalte des Kurznamens sein und keine Konstante (z. B. DOC\_RANK). Der Verweisparameter identifiziert den Kurznamen, auf den die Funktion ES\_SEARCH angewendet werden soll. Der Parameter selbst wird nicht ausgewertet.

Enthält eine Anweisung SELECT in der Klausel FROM mehr als eine Tabelle, und enthält die Klausel WHERE eine Anweisung ES\_SEARCH, kann mit Hilfe des Verweisparameters an DB2 übermittelt werden, zu welcher Tabelle eine bestimmte Suchanweisung gehört. Beispiel:

```

SELECT *
FROM ES_Nickname_1 as N1, ES_Nickname_2 as N2
WHERE ESWRAPPER.ES_SEARCH(N1.DOC_RANK, 'IBM')=1 AND
      ESWRAPPER.ES_SEARCH(N2.DOC_RANK, 'LOTUS')=1

```

**VARCHAR(1024)**

Definiert den Abfrageausdruck. Dieser Parameter muss in einer Abfrage eine Zeichenfolge angeben, die die Abfragesprache GQL von Extended Search verwendet.

**Zugehörige Tasks:**

- „Registrieren der angepassten Funktionen für den Extended Search-Wrapper“ auf Seite 275

**Zugehörige Referenzen:**

- „CREATE FUNCTION (Sourced or Template) statement“ in *SQL Reference, Volume 2*
- „Extended Search-Wrapper - Beispielabfragen“ auf Seite 278
- „Extended Search-Wrapper - Abfragesprache GQL (Generalized Query Language)“ auf Seite 281

---

## Entrez-DDL-Referenzinformationen

### CREATE SERVER-Anweisungsargumente - Entrez-Wrapper

Folgende Argumente sind der Anweisung CREATE SERVER für Entrez zugeordnet:

**TYPE** Gibt den Typ der Datenquelle an. Akzeptable Werte für den Servertyp sind PubMed und Nucleotide. Sie sind von der Groß-/Kleinschreibung abhängig.

**VERSION**

Gibt die Version des von Ihnen verwendeten NCBI-XML-Schemas an. Dieses Argument ist optional. Ist die Version des Servers nicht angegeben, wird der Standardwert 1.0 verwendet.

**WRAPPER**

Gibt den Namen des Wrappers an, den Sie mit der Anweisung CREATE WRAPPER registriert haben.

**Zugehörige Tasks:**

- „Registrieren des Servers für eine Entrez-Datenquelle“ auf Seite 226

**Zugehörige Referenzen:**

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Anhang D, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 641

### CREATE NICKNAME-Anweisungsoptionen - Entrez-Wrapper

Nachfolgend sind die Optionen der Anweisung CREATE NICKNAME für Entrez beschrieben:

**REMOTE\_OBJECT**

Gibt den Namen des Entrez-Objekttyps an, der dem Kurznamen zugeordnet ist. Dieser Name ermittelt das Schema und die NCBI-Datenbank für den Kurznamen und dessen Beziehung zu anderen Kurznamen. Dieser Name ist von der Groß-/Kleinschreibung abhängig.

**PARENT**

Wird nur für einen untergeordneten Kurznamen (Kind) angegeben, dessen übergeordneter Kurzname (Elter) über die Option REMOTE\_OBJECT umbenannt wurde. Die Option PARENT ordnet ein Kind einem Elter zu, wenn innerhalb eines DB2-Schemas mehrere Kurznamenfamilien definiert wurden. Dieser Name ist abhängig von der Groß-/Kleinschreibung.

**Zugehörige Tasks:**

- „Registrieren von Kurznamen für Entrez-Datenquellen“ auf Seite 228

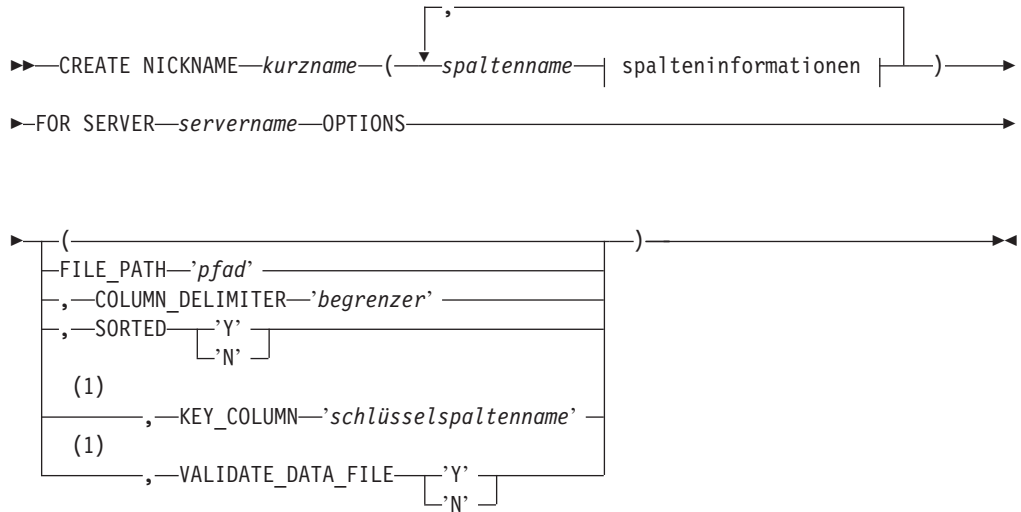
**Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*

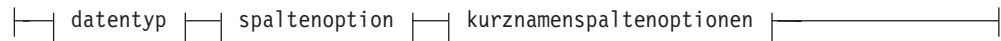
# Referenzinformationen zur DDL von Dateien mit Tabellenstruktur

## Syntax der Anweisung CREATE NICKNAME - Wrapper für Dateien mit Tabellenstruktur

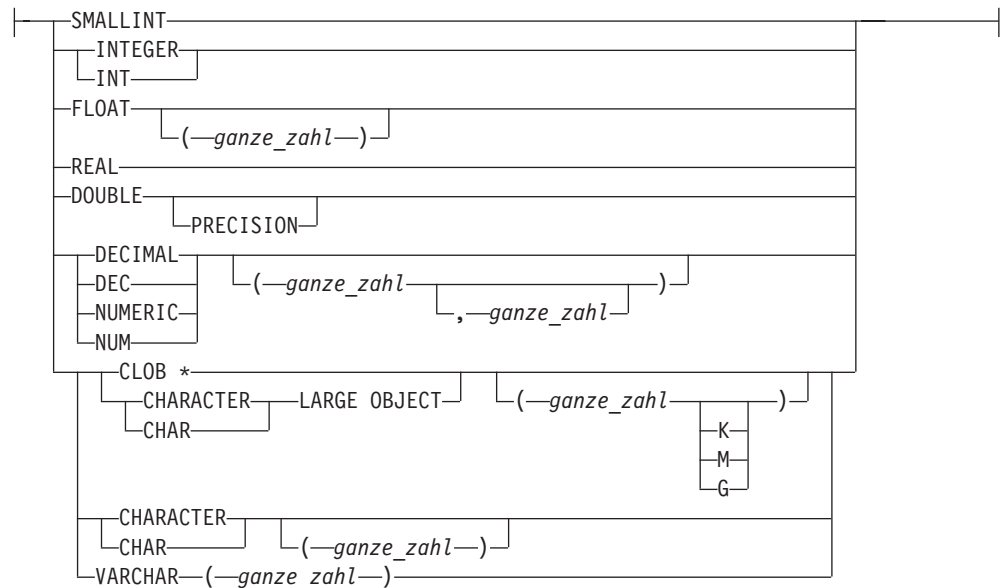
Die Syntax für die Anweisung CREATE NICKNAME sieht wie folgt aus:



### spalteninformationen:



### datentyp:



### spaltenoption:

|-----|  
| NOT NULL |

### kurznamenoptionen:

|-----|  
| OPTIONS (DOCUMENT 'FILE' ) |

### Anmerkungen:

1 Für unsortierte Dateien nicht zulässig. Für sortierte Dateien optional.

**Einschränkung:** Für den Wrapper für Dateien mit Tabellenstruktur ist die Länge von CLOB auf 5 Megabyte (5 MB) beschränkt.

#### *kurzname*

Ein eindeutiger Kurzname für die Datei mit Tabellenstruktur, auf die zugegriffen werden soll. Er muss sich von allen anderen Kurznamen, Tabellen und Sichten in dem Schema unterscheiden, in dem er registriert wird.

#### *spaltenname*

Ein eindeutiger Name, der jedem Feld in der Datei mit Tabellenstruktur gegeben wird. Geben Sie nach jedem Spaltennamen den jeweiligen Datentyp an. Es werden nur Spalten vom Typ CHAR, CLOB, DECIMAL, DOUBLE, FLOAT, INTEGER, REAL, SMALLINT und VARCHAR unterstützt.

### **CHARACTER(*ganze\_zahl*) oder CHAR(*ganze\_zahl*) oder CHARACTER oder CHAR**

Gibt eine Zeichenfolge mit der festen Länge *ganze\_zahl* an, die von 1 bis 254 Zeichen lang sein kann. Bei fehlender Längenangabe wird eine Länge von 1 Zeichen angenommen.

### **CLOB(*ganze\_zahl*)**

Gibt ein großes Zeichenobjekt mit einer Länge von *ganze\_zahl* an, wobei die Länge maximal 5 Megabyte betragen kann. Bei fehlender Längenangabe wird eine Länge von 1 Megabyte angenommen.

### **DECIMAL(*genauigkeitszahl*, *kommastellenzahl*) oder DEC(*genauigkeitszahl*, *kommastellenzahl*)**

Gibt eine Dezimalzahl an.

Die zuerst angegebene ganze Zahl definiert die Genauigkeit der Dezimalzahl, das heißt, die Gesamtanzahl der Stellen. Es kann ein Wert aus dem Bereich von 1 bis 31 angegeben werden.

Die zweite Zahl definiert die Anzahl der Kommastellen der Dezimalzahl, das heißt, die Anzahl von Stellen, die rechts neben dem Dezimalzeichen stehen. Hier kann ein Wert aus dem Bereich von 0 bis zur Anzahl der Genauigkeitsstellen angegeben werden.

Wenn die Genauigkeit und die Anzahl der Kommastellen nicht angegeben werden, gelten die Standardwerte 5,0.

Die Wörter **NUMERIC** und **NUM** können als Synonyme für **DECIMAL** und **DEC** verwendet werden.



**DOUBLE oder DOUBLE PRECISION**

Gibt eine Gleitkommazahl mit doppelter Genauigkeit an.

**FLOAT(*ganze\_zahl*)**

Gibt eine Gleitkommazahl mit einfacher oder doppelter Genauigkeit an, je nach dem Wert von *ganze\_zahl*. Der Wert *ganze\_zahl* muss eine ganze Zahl aus dem Bereich von 1 bis 53 sein. Die Werte 1 bis 24 geben einfache Genauigkeit, die Werte 25 bis 53 doppelte Genauigkeit an.

**INTEGER oder INT**

Gibt eine große ganze Zahl an.

**REAL** Gibt eine Gleitkommazahl mit einfacher Genauigkeit an.

**SMALLINT**

Gibt eine kleine ganze Zahl an.

**VARCHAR(*ganze\_zahl*)**

Gibt eine Zeichenfolge variabler Länge mit der maximalen Länge *ganze\_zahl* an, die von 1 bis 32672 Zeichen lang sein kann.

**NOT NULL**

Verhindert, dass die Spalte Nullwerte enthält.

Der Wrapper setzt die Integritätsbedingung NOT NULL nicht um, während DB2 dies tut. Wenn Sie einen Kurznamen erstellen und eine Spalte mit einer Integritätsbedingung NOT NULL definieren und anschließend eine Zeile mit SELECT auswählen, die einen Nullwert für die Spalte enthält, gibt DB2 einen Fehler SQL0407N aus, der besagt, dass kein Nullwert einer NOT NULL-Spalte zugeordnet werden kann.

Für sortierte Kurznamen gilt eine Ausnahme von dieser Regel. Die Schlüsselspalte für sortierte Kurznamen darf nicht NULL (kein Nullwert) sein. Wenn eine Schlüsselspalte mit einem Nullwert für einen sortierten Kurznamen festgestellt wird, wird der Fehler SQL1822N ausgegeben, der besagt, dass die Schlüsselspalte fehlt.

**FOR SERVER**

Identifiziert den Server, den Sie mit der Anweisung CREATE SERVER registriert haben. Dieser Server wird für den Zugriff auf die Datei mit Tabellenstruktur verwendet.

**FILE\_PATH**

Der vollständig qualifizierte Pfad für die Datei mit Tabellenstruktur, auf die zugegriffen werden soll, eingeschlossen in einfache Anführungszeichen. Die Datendatei muss eine Standarddatei oder eine symbolische Verbindung sein. Sie darf keine Pipe oder ein anderer, vom Standard abweichender Dateityp sein. Es muss entweder die Kurznamenspaltenoption FILE\_PATH oder DOCUMENT angegeben werden. Wird die Kurznamenspaltenoption FILE\_PATH angegeben, darf keine Kurznamenspaltenoption DOCUMENT angegeben werden.

**SORTED**

Gibt an, ob die Datenquellendatei sortiert oder unsortiert ist. Gültige Werte für diese Option sind 'Y', 'y', 'n' oder 'N'. Der Standardwert ist 'N'.

Sortierte Datenquellen müssen in aufsteigender Reihenfolge gemäß der Sortierfolge der aktuellen Ländereinstellung sortiert werden, wie sie durch die Einstellungen in der NLS-Kategorie LC\_COLLATE definiert wird.

Wenn Sie angeben, dass es sich um eine sortierte Datenquelle handelt, wird empfohlen, VALIDATE\_DATA\_FILE auf 'Y' zu setzen.

## **COLUMN\_DELIMITER**

Der Begrenzer, der zur Trennung der Spalten in der Datei mit Tabellenstruktur verwendet wird, eingeschlossen in einfache Anführungszeichen. Der Begrenzer kann aus einem oder mehreren Zeichen bestehen. Wenn kein Begrenzer definiert wird, wird standardmäßig das Komma als Begrenzer angenommen. Ein einfaches Anführungszeichen kann nicht als Begrenzer verwendet werden. Das Spaltenbegrenzungszeichen muss in der gesamten Datei einheitlich sein. Ein Nullwert wird durch zwei aufeinanderfolgende Begrenzer dargestellt; ist das NULL-Feld das letzte Feld in der Zeile, wird ein Nullwert durch einen Begrenzer gefolgt von einem Zeilenendezeichen dargestellt. Das Spaltenbegrenzungszeichen darf nicht als gültiges Datenzeichen für eine Spalte vorkommen. Zum Beispiel kann das Komma als Spaltenbegrenzungszeichen nicht verwendet werden, wenn eine der Spalten Datenwerte mit eingebetteten Kommas enthält.

## **KEY\_COLUMN**

Der Name der Spalte in der Datei, die den Schlüssel bildet, nach dem die Datei sortiert ist, eingeschlossen in einfache Anführungszeichen. Diese Option gilt nur für sortierte Dateien. Eine Spalte, die mit der Kurznamenspaltenoption DOCUMENT ausgewiesen wird, darf nicht als Schlüsselspalte angegeben werden.

Es werden nur Schlüssel aus jeweils einer Spalte unterstützt. Schlüssel aus mehreren Spalten sind nicht zulässig. Der Wert muss dem Namen einer Spalte entsprechen, der in der Anweisung CREATE NICKNAME definiert wurde. Die Spalte muss in aufsteigender Reihenfolge sortiert sein. Wenn der Wert für einen sortierten Kurznamen nicht angegeben wird, wird standardmäßig die erste Spalte in der Datei mit dem Kurznamen verwendet. Es wird empfohlen, die Schlüsselspalte so auszuweisen, dass sie keine Nullwerte enthalten darf, indem der Definition der Spalte in der Kurznamenanweisung die Option NOT NULL hinzugefügt wird.

Diese Option ist abhängig von der Groß-/Kleinschreibung. Allerdings setzt DB2 die Spaltennamen in Großbuchstaben um, sofern die Spalte nicht mit doppelten Anführungszeichen definiert wird.

## **VALIDATE\_DATA\_FILE**

Für sortierte Dateien gibt diese Option an, ob der Wrapper prüfen soll, ob die Schlüsselspalte in aufsteigender Reihenfolge sortiert ist und nach Schlüsseln mit dem Wert NULL sucht. Für diese Option sind nur die Werte 'Y' oder 'N', eingeschlossen in einfache Anführungszeichen, gültig. Die Überprüfung findet einmal zum Zeitpunkt der Registrierung statt. Wenn diese Option nicht angegeben wird, findet keine Überprüfung statt. Diese Option ist nicht zulässig, wenn die Kurznamenspaltenoption DOCUMENT für den Dateipfad verwendet wird.

## **DOCUMENT**

Gibt den Typ der Datei mit Tabellenstruktur an. Derzeit unterstützt dieser Wrapper für diese Option lediglich den Wert FILE. Mit der Option DOCUMENT kann pro Kurzname nur eine Spalte angegeben werden. Die der Option DOCUMENT zugeordnete Spalte muss vom Datentyp VARCHAR oder CHAR sein.

Die Verwendung der Kurznamenspaltenoption DOCUMENT anstelle der Kurznamenoption FILE\_PATH impliziert, dass die Datei, die diesem Kurznamen entspricht, während der Abfrage angegeben wird. Ist für die Option DOCUMENT der Wert "FILE" angegeben, bedeutet dies, dass während der Abfrage der vollständige Pfad der Datei angegeben wird, deren Schema der Kurznamendefinition für diesen Kurznamen entspricht. Das folgende Beispiel der Anweisung CREATE NICKNAME zeigt die Verwendung der Kurznamenspaltenoption DOCUMENT.

```
CREATE NICKNAME customers
(
  doc      VARCHAR(100)  OPTIONS(DOCUMENT 'FILE'),
  name    VARCHAR(16),
  address  VARCHAR(30),
  id      VARCHAR(16)
)
FOR SERVER file_server
```

Die folgende Abfrage, in der die Position der Datei mit Tabellenstruktur in der WHERE-Klausel angegeben wird, kann nun für den Kurznamen 'customers' ausgeführt werden:

```
SELECT name, address, id FROM customers
WHERE doc='/home/db2user/Customers.txt'
```

#### **Zugehörige Tasks:**

- „Registrieren des Servers für Dateien mit Tabellenstruktur“ auf Seite 408

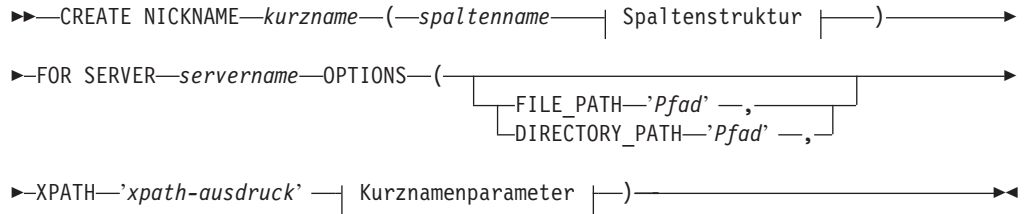
#### **Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den Wrapper für Dateien mit Tabellenstruktur“ auf Seite 409

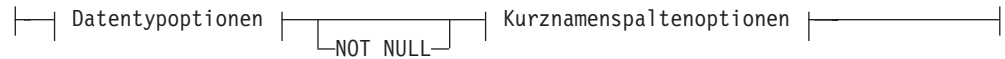
# XML-DDL-Referenzinformationen

## Syntax der Anweisung CREATE NICKNAME - XML-Wrapper

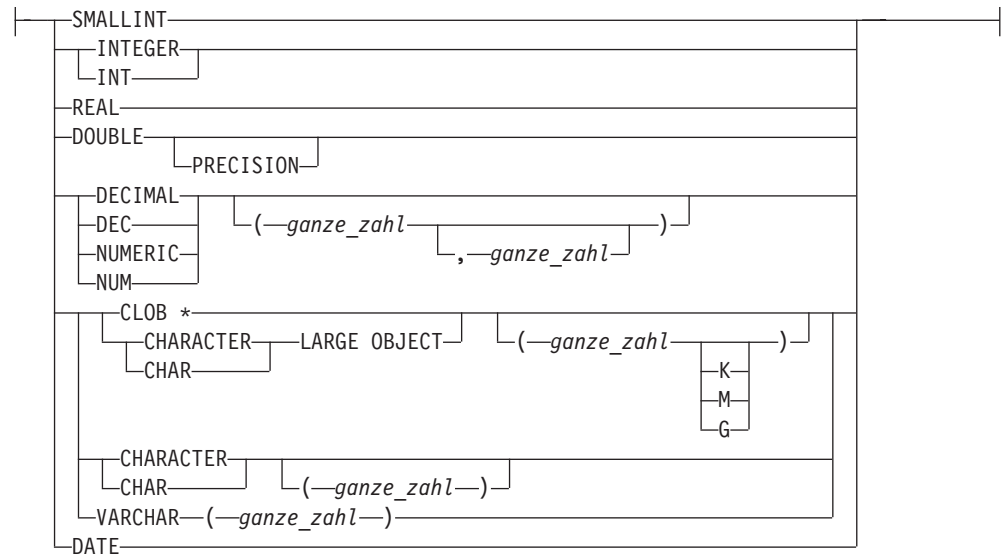
Die Syntax für die Anweisung CREATE NICKNAME sieht wie folgt aus:



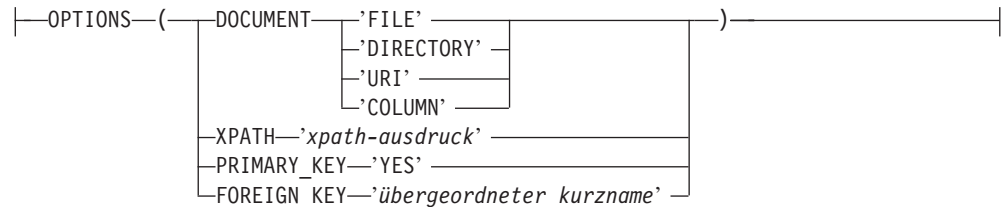
### Spaltenstruktur:



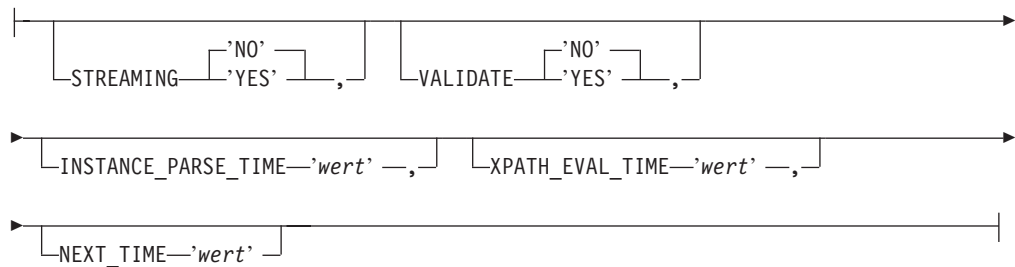
### Datentypoptionen:



### Kurznamenspaltenoptionen:



## Kurznamenparameter:



**Einschränkung:** Für den XML-Wrapper ist die Länge von CLOB auf 5 Megabyte (5 MB) beschränkt.

## Kurznamenparameter und -optionen:

### FILE\_PATH

Gibt den Dateipfad für das XML-Dokument an. Wenn Sie diese Kurznamenoption angeben, dürfen Sie keine DOCUMENT-Spalte angeben. Diese Option FILE\_PATH wird nur für den Stammkurznamen (Root) akzeptiert, d. h. für den Kurznamen, der die Elemente im obersten Teil des XML-Dokuments identifiziert.

### DIRECTORY\_PATH

Gibt den Pfadnamen eines Verzeichnisses an, das mindestens eine XML-Datei enthält. Verwenden Sie diese Option, um einen einzelnen Kurznamen für mehrere XML-Quellendateien zu erstellen. Der XML-Wrapper verwendet ausschließlich Dateien mit der Erweiterung .xml, die sich in dem von Ihnen angegebenen Verzeichnis befinden. Der XML-Wrapper ignoriert alle anderen Dateien in diesem Verzeichnis. Wenn Sie diese Kurznamenoption angeben, dürfen Sie keine DOCUMENT-Spalte angeben. Diese Option DIRECTORY\_PATH wird nur für den Stammkurznamen (Root) akzeptiert, d. h. für den Kurznamen, der die Elemente im obersten Teil des XML-Dokuments identifiziert.

### XPATH

Gibt einen XPath-Ausdruck an, der die XML-Elemente identifiziert, die einzelne Tupel darstellen. Die Kurznamenoption XPATH für einen untergeordneten Kurznamen wird im Kontext des Pfads ausgewertet, der von der Kurznamenoption XPATH des entsprechenden übergeordneten Kurznamens angegeben wird. Dieser XPath-Ausdruck wird als Kontext für die Auswertung der Spaltenwerte verwendet, die von den XPATH-Kurznamenoptionen identifiziert werden.

In einem XPath-Ausdruck können keine Namensbereichspräfixe angegeben werden. Der XML-Wrapper unterstützt keine Namensbereiche.

## Kurznamenspaltenoptionen:

### DOCUMENT

Gibt an, dass es sich bei dieser Spalte um eine DOCUMENT-Spalte handelt. Der Wert der DOCUMENT-Spalte gibt den Typ der XML-Quelldaten an, die dem Kurznamen bei der Ausführung der Abfrage zur Verfügung gestellt werden. Diese Option wird nur für Spalten des Stammkurznamens (Root) akzeptiert, d. h. des Kurznamens, der die Elemente im obersten Teil des XML-Dokuments identifiziert. Mit der Option DOCUMENT kann pro Kurzname nur eine Spalte angegeben werden. Die Spalte, die der Option DOCUMENT zugeordnet ist, muss vom Datentyp VARCHAR sein.

Wenn Sie die Spaltenoption DOCUMENT anstelle der Kurznamenoption FILE\_PATH oder DIRECTORY\_PATH verwenden, wird bei Ausführung der Abfrage das Dokument zur Verfügung gestellt, das diesem Kurznamen entspricht.

Gültige Werte für die Option DOCUMENT sind:

**FILE** Gibt an, dass der Wert der Kurznamenspalte an den Pfadnamen einer Datei gebunden ist, die ein XML-Dokument enthält. Die Daten aus dieser Datei werden beim Ausführen der Abfrage zur Verfügung gestellt.

### DIRECTORY

Gibt an, dass der Wert der Kurznamenspalte an den Pfadnamen eines Verzeichnisses gebunden ist, das mehrere XML-Datendateien enthält. Die XML-Daten aus mehreren Dateien werden beim Ausführen der Abfrage zur Verfügung gestellt. Die Daten befinden sich in XML-Dateien, die unter dem angegebenen Verzeichnispfad gespeichert sind. Der XML-Wrapper verwendet ausschließlich Dateien mit der Erweiterung .xml, die sich in dem von Ihnen angegebenen Verzeichnis befinden. Der XML-Wrapper ignoriert alle anderen Dateien in diesem Verzeichnis.

**URI** Gibt an, dass der Wert der Kurznamenspalte an den Pfadnamen einer fernen XML-Datei gebunden ist, auf die eine URI (Universal Resource Identifier) verweist. Die URI-Adresse gibt die ferne Speicherposition der betreffenden XML-Datei im Web an.

### COLUMN

Gibt an, dass das XML-Dokument in einer relationalen Spalte gespeichert ist.

### XPATH

Gibt den XPath-Ausdruck in dem XML-Dokument an, das die Daten enthält, die dieser Spalte entsprechen. Der XML-Wrapper wertet den XPath-Ausdruck aus, nachdem die Anweisung CREATE NICKNAME diesen XPath-Ausdruck aus dieser XPATH-Kurznamenoption angewendet hat.

Wenn Sie eine Abfrage für einen Spaltennamen ausführen, bei dem ein XPATH-Befehlsverweis nicht richtig konfiguriert ist (wie beispielsweise Fehler bei der Groß-/Kleinschreibung), gibt die Abfrage für alle zurückgegebenen Zeilen in dieser Spalte Nullwerte zurück.

In einem XPath-Ausdruck können keine Namensbereichspräfixe angegeben werden. Der XML-Wrapper unterstützt keine Namensbereiche.

### PRIMARY\_KEY

Gibt an, dass es sich bei diesem Kurznamen um einen übergeordneten Kurznamen (Elter) handelt. Der Spaltendatentyp muss VARCHAR(16) lauten. Ein Kurzname kann höchstens über eine Spaltenoption PRIMARY-

\_KEY verfügen. 'YES' ist der einzige gültige Wert. Die mit dieser Option ausgewiesene Spalte enthält einen vom Wrapper generierten Schlüssel. Der Wert der Spalte kann nicht in einer SELECT-Abfrage abgerufen werden, und die Option XPATH darf für diese Spalte nicht angegeben werden. Die Spalte kann nur zur Verknüpfung von über- und untergeordneten Kurznamen (Eltern und Kindern) verwendet werden.

#### **FOREIGN\_KEY**

Gibt an, dass es sich bei diesem Kurznamen um einen untergeordneten Kurznamen handelt. Der Name des entsprechenden übergeordneten Kurznamens wird ebenfalls angegeben. Ein Kurzname kann höchstens über eine Spaltenoption FOREIGN\_KEY verfügen. Der Wert für diese Option hängt von der Groß-/Kleinschreibung ab. Die mit dieser Option ausgewiesene Spalte enthält einen vom Wrapper generierten Schlüssel. Der Wert der Spalte kann nicht in einer SELECT-Abfrage abgerufen werden, und die Option XPATH darf für diese Spalte nicht angegeben werden. Die Spalte kann nur zur Verknüpfung von über- und untergeordneten Kurznamen (Eltern und Kindern) verwendet werden.

Eine Anweisung CREATE NICKNAME mit einer Option FOREIGN\_KEY schlägt fehl, wenn der übergeordnete Kurzname einen unterschiedlichen Schemanamen hat.

Sofern der Kurzname, auf den in einer FOREIGN\_KEY-Klausel verwiesen wird, nicht explizit für Kleinschreibung oder Groß-/Kleinschreibung definiert wurde, indem er in der entsprechenden Anweisung CREATE NICKNAME in Anführungszeichen gesetzt wurde, muss der Kurzname in Großbuchstaben angegeben werden, wenn Sie in der FOREIGN\_KEY-Klausel auf ihn verweisen.

#### **Kurznamenparameter:**

#### **STREAMING**

Gibt an, ob das XML-Quelldokument in logische Fragmente unterteilt ist, die mit dem Knoten übereinstimmen, der dem XPath-Ausdruck des Kurznamens entspricht. Der XML-Wrapper wird die XML-Quelldaten anschließend Fragment für Fragment syntaktisch analysieren und verarbeiten. Dadurch reduziert sich die Gesamtspeicherbelegung. Sie können STREAMING für alle XML-Quelldokumente angeben (FILE, DIRECTORY, URI oder COLUMN). Diese Option wird nur für Spalten des Stammkurznamens (Root) akzeptiert, d. h. des Kurznamens, der die Elemente im obersten Teil des XML-Dokuments identifiziert. Der Standardwert für STREAMING ist NO.

Der Parameter STREAMING darf nicht auf YES gesetzt werden, wenn der Parameter VALIDATE auf YES gesetzt ist. Werden beide Parameter auf YES gesetzt, empfangen Sie eine Fehlermeldung.

#### **VALIDATE**

Gibt an, ob das XML-Quelldokument vor dem Extrahieren der XML-Daten ausgewertet wird. Wird diese Option auf YES gesetzt, überprüft die Kurznamenoption, ob die Struktur des Quelldokuments mit einem XML-Schema oder einer Dokumenttypdefinition (DTD) übereinstimmt. Diese Option wird nur für Spalten des Stammkurznamens (Root) akzeptiert, d. h. des Kurznamens, der die Elemente im obersten Teil des XML-Dokuments identifiziert. Der Standardwert ist NO.

Das XML-Quelldokument wird nicht geprüft, wenn der XML-Wrapper die XML-Schemadatei oder DTD-Datei (.xsd oder .dtd) nicht finden kann.



DB2 gibt keine Fehlermeldung aus, wenn keine Überprüfung stattfindet. Stellen Sie daher sicher, dass sich die XML-Schemadatei oder DTD-Datei an der Speicherposition befindet, die im XML-Quelldokument angegeben ist. Der Parameter VALIDATE darf nicht auf YES gesetzt werden, wenn der Parameter STREAMING auf YES gesetzt ist. Werden beide Parameter auf YES gesetzt, empfangen Sie eine Fehlermeldung.

#### **INSTANCE\_PARSE\_TIME**

Gibt die Zeit (in Millisekunden) für die Syntaxanalyse der Daten in einer Zeile des XML-Quelldokuments an. Sie können die Optionen INSTANCE\_PARSE\_TIME, XPATH\_EVAL\_TIME und NEXT\_TIME modifizieren, um Abfragen umfangreicher oder komplexer XML-Quellenstrukturen zu optimieren. Diese Option wird nur für Spalten des Stammkurznamens (Root) akzeptiert, d. h. des Kurznamens, der die Elemente im obersten Teil des XML-Dokuments identifiziert. Sie können entweder eine ganze Zahl oder einen Dezimalwert angeben. Der Standardwert ist 7 Millisekunden.

#### **XPATH\_EVAL\_TIME**

Gibt die Zeit (in Millisekunden) für die Auswertung des XPath-Ausdrucks des Kurznamens sowie für die Suche des ersten Elements an. Sie können die Optionen XPATH\_EVAL\_TIME, INSTANCE\_PARSE\_TIME und NEXT\_TIME modifizieren, um Abfragen umfangreicher oder komplexer XML-Quellenstrukturen zu optimieren. Diese Option wird für Stammkurznamen (Root) und Nicht-Stammkurznamen akzeptiert. Sie können entweder eine ganze Zahl oder einen Dezimalwert angeben. Der Standardwert ist 1 Millisekunde.

#### **NEXT\_TIME**

Gibt die erforderliche Zeit (in Millisekunden) für die Suche nachfolgender Quellenelemente im XPath-Ausdruck an. Sie können die Optionen NEXT\_TIME, XPATH\_EVAL\_TIME und INSTANCE\_PARSE\_TIME modifizieren, um Abfragen umfangreicher oder komplexer XML-Quellenstrukturen zu optimieren. Diese Option wird für Stammkurznamen (Root) und Nicht-Stammkurznamen akzeptiert. Der Standardwert ist 1 Millisekunde.

#### **Hinweise zur Verwendung:**

Wenn Sie die Datentypoption DATE verwenden, müssen die Datumsangaben in Ihrem XML-Quelldokument das folgende Format haben: CCJJ-MM-TT. Beispiel: Das Datum 17. November 2002 muss im XML-Quelldokument als 2002-11-17 angegeben werden. Hat ein Datum ein anderes Format, empfangen Sie eine Fehlermeldung. Der Parameter STREAMING darf nicht auf YES gesetzt werden, wenn der Parameter VALIDATE auf YES gesetzt ist und umgekehrt. Der XML-Wrapper überprüft ein gesamtes XML-Quelldokument und keine Quelldokumentfragmente. Werden beide Parameter auf YES gesetzt, empfangen Sie eine Fehlermeldung.

#### **Zugehörige Tasks:**

- „Registrieren von Kurznamen für XML-Datenquellen“ auf Seite 482

#### **Zugehörige Referenzen:**

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Anweisung CREATE NICKNAME - Beispiele für den XML-Wrapper“ auf Seite 482





---

## Anhang A. Sichten in der globalen Katalogtabelle mit Informationen der zusammengeschlossenen Datenbank

Die meisten Katalogsichten in einer zusammengeschlossenen Datenbank entsprechen den Katalogsichten einer beliebigen Datenbank von DB2 für Linux, UNIX und Windows. Es gibt verschiedene eindeutige Sichten mit Informationen, die für ein System zusammengeschlossener Datenbanken relevant sind, wie z. B. die Sicht SYSCAT.WRAPPERS.

Wie in den Handbüchern *DB2 for Linux, UNIX and Windows SQL Reference* von Version 6 und Version 7 angemerkt, besteht auf die Sichten SYSCAT in DB2 Version 8 jetzt nur noch Lesezugriff. Wird für eine Sicht im Schema SYSCAT eine Operation UPDATE oder INSERT abgesetzt, schlägt diese fehl. Es wird empfohlen, den Systemkatalog mit Hilfe der Sichten SYSSTAT zu aktualisieren. Ändern Sie auf die Sicht SYSCAT verweisende Anwendungen, sodass diese stattdessen auf die aktualisierbare Sicht SYSSTAT verweisen.

In der folgenden Tabelle werden die Sichten SYSCAT aufgeführt, die Informationen der zusammengeschlossenen Datenbanken enthalten. Auf diese Sichten besteht Lesezugriff.

*Tabelle 138. In einem System zusammengeschlossener Datenbanken typischerweise verwendete Katalogsichten*

Katalogsichten	Beschreibung
SYSCAT.CHECKS	Enthält von Ihnen definierte Informationen zur Prüfung auf Integritätsbedingung.
SYSCAT.COLCHECKS	Enthält Spalten, auf die durch eine Prüfung auf Integritätsbedingung verwiesen wird.
SYSCAT.COLUMNS	Enthält Spalteninformationen über Datenquellenobjekte (Tabellen und Sichten), für die Sie Kurznamen erstellt haben.
SYSCAT.COLOPTIONS	Enthält Informationen über Spaltenoptionswerte, die Sie für einen Kurznamen definiert haben.
SYSCAT.CONSTDEP	Enthält die Abhängigkeit einer von Ihnen definierten informativen Integritätsbedingung.
SYSCAT.DATATYPES	Enthält Datentypinformationen über lokale integrierte und benutzerdefinierte DB2-Datentypen.
SYSCAT.DBAUTH	Enthält die Datenbankberechtigungen einzelner Benutzern und Gruppen.
SYSCAT.FUNCMAPOPTIONS	Enthält Informationen über Optionswerte, die Sie für eine Funktionszuordnung definiert haben.
SYSCAT.FUNCMAPPINGS	Enthält die Funktionszuordnungen zwischen der zusammengeschlossenen Datenbank und den Datenquellenobjekten.
SYSCAT.INDEXCOLUSE	Enthält Spalten, die an einem Index beteiligt sind.

Tabelle 138. In einem System zusammengeschlossener Datenbanken typischerweise verwendete Katalogsichten (Forts.)

Katalogsichten	Beschreibung
SYSCAT.INDEXES	Enthält Indexspezifikationen für Datenquellenobjekte.
SYSCAT.KEYCOLUSE	Enthält Spalten, die an einem Schlüssel beteiligt sind, der durch eine Integritätsbedingung über eindeutige Schlüssel, Primärschlüssel oder Fremdschlüssel definiert wurde.
SYSCAT.REFERENCES	Enthält Informationen über von Ihnen definierte referenzielle Integritätsbedingungen.
SYSCAT.ROUTINES	Enthält lokale benutzerdefinierte DB2-Funktionen bzw. Funktionsschablonen. Funktionsschablonen werden für die Zuordnung zu einer Datenquellenfunktion verwendet.
SYSCAT.REVTYPEMAPPINGS	Diese Sicht wird nicht verwendet. Alle Datentypzuordnungen werden in der Sicht SYSCAT.TYPEMAPPINGS aufgezeichnet.
SYSCAT.SERVEROPTIONS	Enthält Informationen über Serveroptionswerte, die Sie mit einer Serverdefinition definieren.
SYSCAT.SERVERS	Enthält Serverdefinitionen, die Sie für Datenquellenserver erstellen.
SYSCAT.TABCONST	Jede Zeile stellt eine Tabelle und Integritätsbedingungen für Kurznamen des Typs CHECK, UNIQUE, PRIMARY KEY oder FOREIGN KEY dar.
SYSCAT.TABLES	Enthält Informationen über alle von Ihnen erstellten lokalen DB2-Tabellen, -Sichten für zusammengeschlossene Datenbanken und -Kurznamen.
SYSCAT.TYPEMAPPINGS	Enthält progressive und regressive Datentypzuordnungen. Die Zuordnung erfolgt von Datentypen der Datenquelle zu lokalen DB2-Datentypen. Diese Zuordnungen werden bei der Erstellung eines Kurznamens für ein Datenquellenobjekt verwendet.
SYSCAT.USEROPTIONS	Enthält Informationen über Benutzerberechtigungen, die beim Erstellen von Benutzerzuordnungen zwischen der zusammengeschlossenen Datenbank und den Datenquellenservern eingerichtet werden.
SYSCAT.VIEWS	Enthält Informationen über von Ihnen erstellte lokale Sichten für zusammengeschlossene Datenbanken.
SYSCAT.WRAPOPTIONS	Enthält Informationen über Optionswerte, die Sie für einen Wrapper definiert haben.
SYSCAT.WRAPPERS	Enthält den Namen des Wrappers und der Bibliotheksdatei für jede Datenquelle, für die Sie einen Wrapper erstellen.

In der folgenden Tabelle werden die Sichten SYSSTAT aufgeführt, die Informationen der zusammengeschlossenen Datenbanken enthalten. Auf diese Sichten besteht Lese- und Schreibzugriff. Sie enthalten aktualisierbare Statistiken.

*Tabelle 139. Zusammengeschlossene aktualisierbare globale Katalogsichten*

<b>Katalogsichten</b>	<b>Beschreibung</b>
SYSSTAT.COLUMNS	Enthält statistische Informationen über alle Spalten in Datenquellenobjekten (Tabellen und Sichten), für die Sie Kurznamen erstellt haben. Für aus typisierten Tabellen übernommene Spalten werden keine Statistiken aufgezeichnet.
SYSSTAT.INDEXES	Enthält statistische Informationen über alle Indexspezifikationen für Datenquellenobjekte.
SYSSTAT.ROUTINES	Enthält statistische Informationen über alle benutzerdefinierten Funktionen. Integrierte Funktionen sind darin nicht enthalten. Für aus typisierten Tabellen übernommene Spalten werden keine Statistiken aufgezeichnet.
SYSSTAT.TABLES	Enthält Informationen über alle Basistabellen. Informationen über Sichten, Synonyme und Aliasnamen sind in dieser Sicht nicht enthalten. Bei typisierten Tabellen schließt die Sicht nur die Stammtabelle aus einer Tabellenhierarchie mit ein. Für aus typisierten Tabellen übernommene Spalten werden keine Statistiken aufgezeichnet.



## Anhang B. Wrapperoptionen für Systeme zusammengeschlusener Datenbanken

Wrapperoptionen werden zur Konfiguration des Wrappers verwendet. Sie dienen aber auch dazu, festzulegen, wie der Wrapper vom Server mit zusammengeschlusenen Datenbanken verwendet wird. Wrapperoptionen können bei der Erstellung oder Änderung des Wrappers definiert werden.

Alle relationalen und nicht relationalen Datenquellen verwenden die Wrapperoption DB2\_FENCED. Die ODBC-Datenquelle verwendet die Wrapperoption MODULE. Die Entrez-Datenquelle verwendet die Wrapperoption EMAIL.

Tabelle 140. Wrapperoptionen und deren Einstellungen

Option	Gültige Einstellungen	Standardeinstellung
DB2_FENCED	Gibt an, ob der Wrapper im abgeschirmten bzw. im gesicherten Modus ausgeführt wird.	Relationale Wrapper: N. Nicht relationale IBM Wrapper: N.
	Y Der Wrapper wird im abgeschirmten Modus ausgeführt.	Nicht relationale Wrapper von Fremdanbietern: Y.
	N Der Wrapper wird im gesicherten Modus ausgeführt.	
EMAIL	Gibt eine E-Mail-Adresse an, wenn Sie den Entrez-Wrapper registrieren. Diese E-Mail-Adresse ist in allen Abfragen enthalten. Über sie kann NCBI im Problemfall Kontakt zu Ihnen aufnehmen, wenn z. B. zu viele Abfragen die NCBI-Server überlasten. Diese Option ist erforderlich.	
MODULE	Gibt den vollständigen Pfad der Bibliothek an, die die Implementierung von ODBC Driver Manager oder die SQL/CLI-Implementierung enthält. Für den ODBC-Wrapper auf UNIX-Servern mit zusammengeschlusenen Datenbanken erforderlich.	Für Windows lautet der Standardwert odbc32.dll.

### Zugehörige Konzepte:

- „Parallelität mit Abfragen, die auf Kurznamen verweisen“ in *Systeme zusammengeschlusener Datenbanken*

### Zugehörige Tasks:

- „Trusted and fenced mode process environments“ in *IBM DB2 Information Integrator Wrapper Developer's Guide*
- „Ändern einer Oberfläche“ in *Systeme zusammengeschlusener Datenbanken*
- „Registrieren von Wrappern für eine Datenquelle“ auf Seite 70



---

## Anhang C. Gültige Servertypen in SQL-Anweisungen

Servertypen geben an, welchen Datenquellentyp die Serverdefinition darstellt. Servertypen sind je nach Lieferant, Zweck und Betriebssystem unterschiedlich. Unterstützte Werte sind vom verwendeten Wrapper abhängig.

Bei den meisten Datenquellen müssen Sie in der Anweisung CREATE SERVER einen gültigen Servertyp angeben.

---

### BioRS-Wrapper

BioRS-Datenquellen.

Servertyp	Datenquelle
In der Anweisung CREATE SERVER nicht erforderlich.	BioRS

---

### BLAST-Wrapper

Vom BLAST-Dämon unterstützte BLAST-Datenquellen.

Servertyp	Datenquelle
BLASTN	BLAST-Suchoperationen, in denen eine Nukleotidsequenz mit dem Inhalt einer Nukleotidsequenzdatenbank verglichen wird, um Sequenzen zu finden, die Regionen enthalten, die Homologien mit Regionen in der ursprünglichen Sequenz aufweisen.
BLASTP	BLAST-Suchoperationen, in denen eine Aminosäuresequenz mit dem Inhalt einer Aminosäuresequenzdatenbank verglichen wird, um Sequenzen zu finden, die Regionen enthalten, die Homologien mit Regionen in der ursprünglichen Sequenz aufweisen.
BLASTX	BLAST-Suchoperationen, in denen eine Nukleotidsequenz mit dem Inhalt einer Aminosäuresequenzdatenbank verglichen wird, um Sequenzen zu finden, die Regionen enthalten, die Homologien mit Regionen in der ursprünglichen Sequenz aufweisen.
TBLASTN	BLAST-Suchoperationen, in denen eine Aminosäuresequenz mit dem Inhalt einer Nukleotidsequenzdatenbank verglichen wird, um Sequenzen zu finden, die Regionen enthalten, die Homologien mit Regionen in der ursprünglichen Sequenz aufweisen.
TBLASTX	BLAST-Suchoperationen, in denen eine Nukleotidsequenz mit dem Inhalt einer Nukleotidsequenzdatenbank verglichen wird, um Sequenzen zu finden, die Regionen enthalten, die Homologien mit Regionen in der ursprünglichen Sequenz aufweisen.



---

## CTLIB-Wrapper

Von der CTLIB-Client-Software unterstützte Sybase-Datenquellen.

Servertyp	Datenquelle
SYBASE	Sybase

---

## Documentum-Wrapper

Von der Documentum-Client-API/-Bibliothek unterstützte Documentum-Datenquellen.

Servertyp	Datenquelle
DCTM	Documentum

---

## DRDA-Wrapper

### Datenquellen der DB2-Familie

*Tabelle 141. DB2 für Linux, UNIX und Windows*

Servertyp	Datenquelle
DB2/UDB	IBM DB2 Universal Database
DB2/6000	IBM DB2 für AIX
DB2/AIX	IBM DB2 für AIX
DB2/HPUX	IBM DB2 für HP-UX
DB2/HP	IBM DB2 für HP-UX
DB2/NT	IBM DB2 für Windows NT
DB2/EEE	IBM DB2 Enterprise-Extended Edition
DB2/SUN	IBM DB2 für Solaris
DB2/PE	IBM DB2 für Personal Edition
DB2/2	IBM DB2 für OS/2
DB2/LINUX	IBM DB2 für Linux
DB2/PTX	IBM DB2 für NUMA-Q
DB2/SCO	IBM DB2 für SCO Unixware

*Tabelle 142. DB2 für iSeries (und AS/400)*

Servertyp	Datenquelle
DB2/400	IBM DB2 für iSeries und AS/400

*Tabelle 143. DB2 für z/OS und OS/390*

Servertyp	Datenquelle
DB2/ZOS	IBM DB2 für z/OS
DB2/390	IBM DB2 für OS/390

Tabelle 143. DB2 für z/OS und OS/390 (Forts.)

Servertyp	Datenquelle
DB2/MVS	IBM DB2 für MVS

Tabelle 144. DB2 Server für VM und VSE

Servertyp	Datenquelle
DB2/VM	IBM DB2 für VM
DB2/VSE	IBM DB2 für VSE
SQL/DS	IBM SQL/DS

## Entrez-Wrapper

Entrez-Datenquellen.

Servertyp	Datenquelle
NUCLEOTIDE	Entrez
PUBMED	Entrez

## Excel-Wrapper

Von Microsoft Excel 97, 2000 und 2002 unterstützte Excel-Datenquellen.

Servertyp	Datenquelle
In der Anweisung CREATE SERVER nicht erforderlich.	Microsoft Excel

## Extended Search-Wrapper

Von der Extended Search Client Library unterstützte Extended Search-Datenquellen.

Servertyp	Datenquelle
In der Anweisung CREATE SERVER nicht erforderlich.	IBM Lotus Extended Search

## HMMER-Wrapper

Vom HMMER-Dämon unterstützte HMMER-Datenquellen.

Servertyp	Datenquelle
PFAM	HMMER
SEARCH	HMMER

---

## Informix-Wrapper

Von der Software Informix Client SDK unterstützte Datenquellen von Informix.

Servertyp	Datenquelle
INFORMIX	Informix

---

## MSSQLODBC3-Wrapper

Von dem Treiber für DataDirect Connect ODBC 3.6 oder ODBC 3.0 (oder höher) unterstützte Microsoft SQL Server-Datenquellen.

Servertyp	Datenquelle
MSSQLSERVER	Microsoft SQL Server

---

## NET8-Wrapper

Von der Oracle NET8-Client-Software unterstützte Datenquellen von Oracle.

Servertyp	Datenquelle
ORACLE	Oracle Version 8.0. oder höher

---

## ODBC-Wrapper

Von dem ODBC 3.x-Treiber unterstützte Datenquellen von ODBC.

Servertyp	Datenquelle
ODBC	ODBC

---

## OLE DB-Wrapper

Mit Microsoft OLE DB 2.0 oder höher kompatible OLE DB-Provider.

Servertyp	Datenquelle
In der Anweisung CREATE SERVER nicht erforderlich.	Beliebiger OLE DB-Provider

---

## Wrapper für Dateien mit Tabellenstruktur

Als Datenquellen verwendete Dateien mit Tabellenstruktur.

Servertyp	Datenquelle
In der Anweisung CREATE SERVER nicht erforderlich.	Dateien mit Tabellenstruktur

---

---

## Teradata-Wrapper

Von der Teradata V2R3-, V2R4- und V2R5-Client-Software unterstützte Datenquellen von Teradata.

Servertyp	Datenquelle
TERADATA	Teradata

---

## Wrapper für Web-Services

Web-Services-Datenquellen.

Servertyp	Datenquelle
In der Anweisung CREATE SERVER nicht erforderlich.	Beliebige Web-Services-Datenquelle.

---

## WebSphere Business Integration-Wrapper

Als Datenquellen verwendete und vom WebSphere Business Integration-Wrapper unterstützte Geschäftsanwendungen.

Servertyp	Datenquelle
WBI	WebSphere Business Integration 2.2 oder 2.3

---

## XML-Wrapper

XML-Datenquellen.

Servertyp	Datenquelle
In der Anweisung CREATE SERVER nicht erforderlich.	XML



## Anhang D. Serveroptionen für Systeme zusammengeschlusener Datenbanken

Serveroptionen werden verwendet, um den Datenquellenserver zu beschreiben. In Serveroptionen werden Informationen über Datenintegrität, Speicherposition, Sicherheitsbedingungen und Leistung angegeben. Einige der Serveroptionen stehen für alle Datenquellen zur Verfügung, andere Serveroptionen sind datenquellen-spezifisch.

Für Server mit zusammengeschlusenen Datenbanken sind folgende, allgemeine Serveroptionen verfügbar:

- Optionen für die Kompatibilität. COLLATING\_SEQUENCE, IGNORE\_UDT
- Optionen für die Datenintegrität. IUD\_APP\_SVPT\_ENFORCE
- Optionen für Datum und Zeit. DATEFORMAT, TIMEFORMAT, TIMESTAMPFORMAT
- Optionen für die Speicherposition. CONNECTSTRING, DBNAME, IFILE
- Optionen für die Sicherheitsbedingungen. FOLD\_ID, FOLD\_PW, INFORMIX\_LOCK\_MODE
- Optionen für die Leistung. COMM\_RATE, CPU\_RATIO, DB2\_MAXIMAL\_PUSHDOWN, IO\_RATIO, LOGIN\_TIMEOUT, PACKET\_SIZE, PLAN\_HINTS, PUSHDOWN, TIMEOUT, VARCHAR\_NO\_TRAILING\_BLANKS

In der folgenden Tabelle sind die auf die einzelnen relationalen Datenquellen anwendbaren Serveroptionen für die Serverdefinition aufgelistet.

Tabelle 145. Serveroptionen für relationale Datenquellen

Datenquelle	CODEPAGE	COLLATING_SEQUENCE	COMM_RATE	CONNECTSTRING	CPU_RATIO	DATEFORMAT	DB2_MAXIMAL_PUSHDOWN	DBNAME	FOLD_ID	FOLD_PW	IFILE	INFORMIX_LOCK_MODE	IO_RATIO	IUD_APP_SVPT_ENFORCE	LOGIN_TIMEOUT	NODE	PACKET_SIZE	PASSWORD	PLAN_HINTS	PUSHDOWN	TIMEOUT	TIMEFORMAT	TIMESTAMPFORMAT	VARCHAR_NO_TRAILING_BLANKS
DB2 UDB für iSeries		X	X		X		X	X	X	X			X	X				X		X				X
DB2 UDB für z/OS und OS/390		X	X		X		X	X	X	X			X	X				X		X				X
DB2 für VM und VSE		X	X		X		X	X	X	X			X	X				X		X				X

Tabelle 145. Serveroptionen für relationale Datenquellen (Forts.)

Datenquelle	CODEPAGE	COLLATING_SEQUENCE	COMM_RATE	CONNECTSTRING	CPU_RATIO	DATEFORMAT	DB2_MAXIMAL_PUSHDOWN	DBNAME	FOLD_ID	FOLD_PW	IFILE	INFORMIX_LOCK_MODE	IO_RATIO	IUD_APP_SVPT_ENFORCE	LOGIN_TIMEOUT	NODE	PACKET_SIZE	PASSWORD	PLAN_HINTS	PUSHDOWN	TIMEOUT	TIMEFORMAT	TIMESTAMPFORMAT	VARCHAR_NO_TRAILING_BLANKS
DB2 UDB für Linux, UNIX und Windows		X	X		X		X	X	X	X			X	X				X		X				X
Informix		X	X		X		X	X	X	X		X	X	X		X		X		X				
Microsoft SQL Server	X	X	X		X		X	X	X	X			X	X		X		X		X				
ODBC	X	X	X		X	X	X	X	X	X			X	X		X		X		X		X	X	X
OLE DB		X		X																				
Oracle		X	X		X		X		X	X			X			X		X	X	X				X
Sybase		X	X		X		X	X	X	X	X		X		X	X	X	X	X	X	X			
Teradata		X	X		X		X						X	X		X				X				

In der folgenden Tabelle sind die für die einzelnen nicht relationalen Datenquellen (mit Ausnahme von WebSphere Business Integration) geltenden Serveroptionen für die Serverdefinition aufgelistet. Die Serveroptionen für die Serverdefinition, die für WebSphere Business Integration verfügbar sind, werden in Tabelle 147 auf Seite 643 aufgelistet.

Tabelle 146. Serveroptionen für nicht relationale Datenquellen

Datenquelle	CASE_SENSITIVE	CONTENT_DIR	DAEMON_PORT	ES_HOST	ES_PORT	ES_TRACING	ES_TRACELEVEL	ES_TRACEFILENAME	HMMPEM_OPTIONS	HMMSEARCH_OPTIONS	MAX_ROWS	NODE	OS_TYPE	PORT	PROCESSORS	PROXU_AUTHID	PROXY_PASSWORD	PROXY_SERVER_NAME	PROXY_SERVER_PORT	PROXY_TYPE	RDBMS_TYPE	SOCKET_TIMEOUT	TIMEOUT	TRANSACTIONS	USE_CLOB_SEQUENCE
BioRS	X											X		X									X		
BLAST			X									X													X
Documentum		X										X	X											X	
Entrez											X					X	X	X	X	X		X			
Excel																									

Tabelle 146. Serveroptionen für nicht relationale Datenquellen (Forts.)

Datenquelle	CASE_SENSITIVE	CONTENT_DIR	DAEMON_PORT	ES_HOST	ES_PORT	ES_TRACING	ES_TRACELEVEL	ES_TRACEFILENAME	HMMPFAM_OPTIONS	HMMSEARCH_OPTIONS	MAX_ROWS	NODE	OS_TYPE	PORT	PROCESSORS	PROXU_AUTHID	PROXY_PASSWORD	PROXY_SERVER_NAME	PROXY_SERVER_PORT	PROXY_TYPE	RDBMS_TYPE	SOCKET_TIMEOUT	TIMEOUT	TRANSACTIONS	USE_CLOB_SEQUENCE
Extended Search				X	X	X	X	X																	
HMMER			X						X	X		X			X										X
Dateien mit Tabellenstruktur																									
Web-Services																									
XML																X	X	X	X	X		X			

In der folgenden Tabelle sind die Serveroptionen für die Serverdefinition aufgelistet, die für WebSphere Business Integration-Datenquellen verwendet werden können.

Tabelle 147. Serveroptionen für WebSphere Business Integration-Datenquellen

Datenquelle	APP_TYPE	FAULT_QUEUE	MQ_CONN_NAME	MQ_MANAGER	MQ_RESPONSE_TIMEOUT	MQ_SVRCONN_CHANNELNAME	REQUEST_QUEUE	RESPONSE_QUEUE
WebSphere Business Integration	X	X	X	X	X	X	X	X

In der folgenden Tabelle sind die einzelnen Serveroptionen beschrieben und die gültigen Einstellungen sowie die jeweilige Standardeinstellung aufgeführt.

Tabelle 148. Serveroptionen und deren Einstellungen

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
APP_TYPE	Der Typ der fernen Anwendung. Zulässige Werte sind 'PSOFT', 'SAP' und 'SIEBEL'. Diese Option ist erforderlich.	keine



Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
CASE_SENSITIVE	<p>Gibt an, ob der BioRS-Server bei der Verarbeitung von Namen die Groß-/Kleinschreibung berücksichtigt. Zulässige Werte sind Y oder N.</p> <p>'Y' Der BioRS-Server verarbeitet Namen unter Berücksichtigung der Groß-/Kleinschreibung.</p> <p>'N' Der BioRS-Server verarbeitet Namen ohne Berücksichtigung der Groß-/Kleinschreibung.</p>	Y
	<p>Im BioRS-Produkt wird die Beachtung der Groß-/Kleinschreibung bei den auf dem BioRS-Server gespeicherten Daten über einen entsprechenden Konfigurationsparameter gesteuert. Die Option CASE_SENSITIVE stellt die DB2 Information Integrator-Entsprechung zu diesen BioRS-Systemkonfigurationsparameter dar. Sie müssen die Konfigurationseinstellungen des BioRS-Servers für die Beachtung der Groß-/Kleinschreibung zwischen Ihrem BioRS-System und DB2 Information Integrator synchronisieren. Andernfalls kommt es zu Fehlern, wenn Sie versuchen, über DB2 Information Integrator auf BioRS-Daten zuzugreifen.</p>	
	<p>Nach dem Erstellen eines neuen BioRS-Servers in DB2 Information Integrator kann die Option CASE_SENSITIVE nicht mehr geändert oder gelöscht werden. Muss die Option CASE_SENSITIVE dennoch geändert werden, müssen Sie den gesamten Server nochmals löschen und dann erneut erstellen. Wenn Sie den BioRS-Server löschen, müssen auch alle zugehörigen BioRS-Kurznamen neu erstellt werden. DB2 Information Integrator löscht automatisch alle Kurznamen, die einem gelöschten Server zugeordnet sind.</p>	
CODEPAGE	<p>Gibt die DB2-Codepagekennung an, die dem codierten Zeichensatz der Clientkonfiguration für die Datenquelle entspricht. Sie müssen die Client-Codepage angeben, wenn diese und die Codepage der zusammengeschlossenen Datenbank nicht übereinstimmen.</p> <p>Bei Datenquellen mit Unicode-Unterstützung kann für die Option CODEPAGE die DB2-Codepagekennung angegeben werden, die der unterstützten Unicode-Codierung des Datenquellenclients entspricht.</p>	<p>Auf UNIX- oder Windows-Systemen mit einer zusammengeschlossenen Datenbank ohne Unicode-Unterstützung: Codepage der zusammengeschlossenen Datenbank.</p> <p>Auf UNIX-Systemen mit einer zusammengeschlossenen Datenbank mit Unicode-Unterstützung: 1208.</p> <p>Auf Windows-Systemen mit einer zusammengeschlossenen Datenbank mit Unicode-Unterstützung: 1202.</p>

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
COLLATING_SEQUENCE	<p>Gibt an, ob die Datenquelle auf der Grundlage des codierten NLS-Zeichensatzes und der Landes-/Regionalinformationen die gleiche Standardsortierfolge verwendet wie die zusammengeslossene Datenbank.</p> <p>'Y' Die Datenquelle verwendet die gleiche Sortierfolge wie die zusammengeslossene DB2-Datenbank.</p> <p>'N' Die Datenquelle verwendet eine andere Sortierfolge als die zusammengeslossene DB2-Datenbank.</p> <p>'I' Die Datenquelle verwendet eine andere Sortierfolge als die zusammengeslossene DB2-Datenbank, wobei bei der Sortierfolge der Datenquelle die Groß-/Kleinschreibung nicht berücksichtigt wird ('STEWART' und 'StewART' werden z. B. gleich behandelt).</p>	'N'
COMM_RATE	<p>Gibt die Übertragungsgeschwindigkeit zwischen dem Server mit zusammengesetzten Datenbanken und dem Datenquellenserver an. Diese Geschwindigkeit wird in Megabyte pro Sekunde angegeben.</p> <p>Die gültigen Werte liegen im Bereich zwischen größer als 0 und <math>1 \times 10^{23}</math>. Die Werte können in jeder zulässigen REAL-Notation definiert werden.</p>	'2'
CONTENT_DIR	Gibt den Namen des Stammverzeichnisses mit lokalem Zugriff an, in dem Nachrichtenkomponenten gespeichert werden, die über die Pseudospalten GET_FILE, GET_FILE_DEL, GET_RENDITION und GET_RENDITION_DEL abgerufen werden. Alle Benutzer, die mit diesen Pseudospalten arbeiten können, müssen über Schreibberechtigung für dieses Verzeichnis verfügen.	<p>Auf UNIX-Systemen: '/tmp'</p> <p>Auf Windows-Systemen: 'C:\temp'</p>
CONNECTSTRING	Gibt die für eine Verbindung zu einem OLE-DB-Provider erforderlichen Merkmale für die Initialisierung an.	keine

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
CPU_RATIO	<p>Gibt den Geschwindigkeitsunterschied zwischen der CPU einer Datenquelle und der CPU des Servers mit zusammengeschlossenen Datenbanken an.</p> <p>Die gültigen Werte liegen im Bereich zwischen größer als 0 und <math>1 \times 10^{23}</math>. Die Werte können in jeder zulässigen REAL-Notation definiert werden.</p> <p>Die Einstellung 1 gibt an, dass die CPU des DB2-Servers mit zusammengeschlossenen Datenbanken und die CPU der Datenquelle die gleiche Geschwindigkeit haben, also ein Verhältnis von 1:1. Die Einstellung .5 gibt an, dass die Geschwindigkeit der CPU des DB2-Servers mit zusammengeschlossenen Datenbanken um 50% unter der Geschwindigkeit der CPU der Datenquelle liegt. Die Einstellung 2 gibt an, dass die Geschwindigkeit der CPU des DB2-Servers mit zusammengeschlossenen Datenbanken doppelt so hoch ist wie die Geschwindigkeit der CPU der Datenquelle.</p>	'1.0'
DATEFORMAT	<p>Das von der Datenquelle verwendete Datumsformat. Verwenden Sie bei der Eingabe des Formats die Werte 'TT', 'MM' und 'JJ' bzw. 'JJJJ', um das numerische Format für das Datum darzustellen. Sie sollten außerdem den Begrenzer, wie z. B. ein Leerzeichen oder Komma, angeben. Zur Darstellung des Datumsformats für '2003-01-01' beispielsweise verwenden Sie 'JJJJ-MM-TT'. Für dieses Feld ist die Dateneingabe optional.</p>	keine
DAEMON_PORT	<p>Gibt die Nummer des Ports an, an dem der Dämon für BLAST- und HMMER-Jobanforderungen empfangsbereit ist. Die Portnummer muss mit der Nummer übereinstimmen, die in der Option DAEMON_PORT der Dämonkonfigurationsdatei angegeben wurde.</p>	BLAST: '4007'; HMMER: '4098'

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
DB2_MAXIMAL_PUSHDOWN	<p>Gibt die primären Bedingungen an, die das Abfrageoptimierungsprogramm bei der Auswahl eines Zugriffsplans verwendet. Das Abfrageoptimierungsprogramm kann Zugriffspläne entweder auf Basis des Aufwands auswählen oder auf Basis der Benutzerbestimmung, dass von den ferneren Datenquellen so viel Abfrageverarbeitung wie möglich übernommen wird.</p> <p>'Y' Das Abfrageoptimierungsprogramm wählt einen Zugriffsplan aus, der mehr Abfrageoperationen im Pushdown-Modus an die Datenquelle weiterleitet als andere Pläne. Wenn mehrere Pläne im gleichen Umfang Abfrageoperationen im Pushdown-Modus weiterleiten, wählt das Abfrageoptimierungsprogramm aus diesen Plänen den Plan mit dem geringsten Aufwand aus.</p> <p>Wenn eine gespeicherte Abfragetabelle (MQT, Materialized Query Table) auf dem Server mit zusammengeschlossenen Datenbanken eine Abfrage teilweise oder vollständig verarbeiten kann, wird ein Zugriffsplan verwendet, der die gespeicherte Abfragetabelle enthält. Die zusammengeschlossene Datenbank leitet Abfragen, die ein kartesisches Produkt ergeben, nicht im Pushdown-Modus weiter.</p> <p>'N' Das Abfrageoptimierungsprogramm wählt einen Zugriffsplan auf Basis des anfallenden Aufwands aus.</p>	'N'
DBNAME	Name der Datenbank der Datenquelle, auf die der Server mit zusammengeschlossenen Datenbanken zugreifen soll. Bei DB2-Datenbanken bezieht sich dieser Wert auf eine bestimmte Datenbank, die für den Aufbau der ersten Verbindung zu einer fernen DB2-Datenbank erforderlich ist. Diese spezielle Datenbank stellt einen Aliasnamen für die ferne DB2-Datenbank dar, die auf dem Server mit zusammengeschlossenen Datenbanken mit Hilfe des Befehls CATALOG DATABASE oder des DB2-Konfigurationsassistenten katalogisiert wird. Dies gilt nicht für Oracle-Datenquellen, da Exemplare von Oracle nur eine Datenbank enthalten.	keine
ES_HOST	Gibt den vollständig qualifizierten Hostnamen bzw. die IP-Adresse des Extended Search-Servers an, den Sie durchsuchen wollen. Diese Option ist erforderlich.	keine
ES_PORT	Gibt die Nummer des Ports an, an dem dieser Extended Search-Server für Anforderungen empfangsbereit ist. Diese Option ist optional.	'6001'

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
ES_TRACING	Gibt an, ob für Fehlermeldungen, Warnungen und Informationsnachrichten, die vom fernen Extended Search-Server generiert werden, die Tracefunktion aktiviert werden soll. Gültige Werte sind:  'OFF' Es werden keine Tracenachrichten protokolliert.  'ON' Es werden Tracenachrichten protokolliert. Diese Option ist optional.	'OFF'
ES_TRACELEVEL	Wenn die Tracefunktion aktiviert ist, gibt diese Option die Nachrichtentypen an, die in der Protokolldatei aufgezeichnet werden. Sie können die folgenden Tracestufen unabhängig voneinander aktivieren und inaktivieren.  'C' Nachrichten zu kritischen Fehlern. 'N' Nachrichten zu nicht kritischen Fehlern. 'W' Warnungen. 'I' Informationsnachrichten. Beispiel: ES_TRACELEVEL 'W' ES_TRACELEVEL 'CN'  Diese Option ist optional.	'C'
ES_TRACEFILENAME	Wenn die Tracefunktion aktiviert ist, gibt diese Option den Namen eines Verzeichnisses und einer Datei an, in dem bzw. in der Nachrichten aufgezeichnet werden. Diese Option ist optional.	Für UNIX-Betriebssysteme: \$INSTHOME/sql/lib/log/ESWrapper.log.  Für Windows-Betriebssysteme: %DB2TEMPDIR%\ESWrapper.log.
FAULT_QUEUE	Der Name der Fehlerwarteschlange, die Fehlermeldungen vom Adapter zum Wrapper weiterleitet. Der Name muss mit den Spezifikationen für die Angabe von Warteschlangennamen in WebSphere MQ übereinstimmen. Diese Option ist erforderlich.	keine

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
FOLD_ID  (Siehe Anmerkungen 1 und 4 am Ende dieser Tabelle.)	<p>Gilt für Benutzer-IDs, die vom Server mit zusammengeschlossenen Datenbanken zur Authentifizierung an den Server der Datenquelle gesendet werden. Gültige Werte sind:</p> <p>'U' Der Server mit zusammengeschlossenen Datenbanken setzt die Benutzer-ID vor dem Senden an die Datenquelle in Großbuchstaben um. Dies ist für Datenquellen der DB2-Produktfamilie und für Oracle-Datenquellen erforderlich (siehe Anmerkung 2 am Ende dieser Tabelle).</p> <p>'N' Der Server mit zusammengeschlossenen Datenbanken verändert die Benutzer-ID vor dem Senden an die Datenquelle nicht. (Siehe Anmerkung 2 am Ende dieser Tabelle.)</p> <p>'L' Der Server mit zusammengeschlossenen Datenbanken setzt die Benutzer-ID vor dem Senden an die Datenquelle in Kleinbuchstaben um.</p> <p>Wird keine dieser Einstellungen verwendet, versucht der Server mit zusammengeschlossenen Datenbanken, die Benutzer-ID in Großschreibung an die Datenquelle zu senden. Schlägt dieser Versuch fehl, versucht der Server, die Benutzer-ID in Kleinschreibung zu senden.</p>	keine
FOLD_PW  (Siehe Anmerkungen 1, 3 und 4 am Ende dieser Tabelle.)	<p>Gilt für Kennwörter, die der Server mit zusammengeschlossenen Datenbanken zur Authentifizierung an Datenquellen sendet. Gültige Werte sind:</p> <p>'U' Der Server mit zusammengeschlossenen Datenbanken setzt das Kennwort vor dem Senden an die Datenquelle in Großbuchstaben um. Dies ist für Datenquellen der DB2-Produktfamilie und für Oracle-Datenquellen erforderlich.</p> <p>'N' Der Server mit zusammengeschlossenen Datenbanken verändert das Kennwort vor dem Senden an die Datenquelle nicht.</p> <p>'L' Der Server mit zusammengeschlossenen Datenbanken setzt das Kennwort vor dem Senden an die Datenquelle in Kleinbuchstaben um.</p> <p>Wird keine dieser Einstellungen verwendet, versucht der Server mit zusammengeschlossenen Datenbanken, das Kennwort in Großschreibung an die Datenquelle zu senden. Schlägt dieser Versuch fehl, versucht der Server, das Kennwort in Kleinschreibung zu senden.</p>	keine

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
HMMPFAM_OPTIONS	<p>Gibt hmmpfam-Optionen wie z. B. --null2, --pvm und --xnu an, die keinen zugehörigen Spaltennamen in einer Referenztabelle haben, mit dem Optionen bestimmten Spaltennamen zugeordnet werden können.</p> <p>Beispiel: HMMPFAM_OPTIONS '--xnu --pvm'</p> <p>In diesem Beispiel führt der Dämon das Programm HMMPFAM mit Optionen aus der WHERE-Klausel der Abfrage sowie mit den zusätzlichen Optionen --xnu --pvm aus.</p>	
HMMSEARCH_OPTIONS	Ermöglicht dem Benutzer die Angabe zusätzlicher Befehlszeilenoptionen im Befehl hmmsearch. Diese Option ist nur beim Typ SEARCH zulässig. Weitere Informationen hierzu finden Sie im HMMER-Benutzerhandbuch.	keine
IFILE	Gibt den Pfad und Namen der Schnittstellendatei für Sybase Open Client an. Für Server mit zusammengeschlossenen Datenbanken unter Windows NT lautet der Standardwert %DB2PATH%\interfaces. Für Server mit zusammengeschlossenen Datenbanken unter UNIX lautet der Standardwert für Pfad und Name \$DB2INSTANCE/sql1lib/interfaces.	keine
INFORMIX_LOCK_MODE	<p>Gibt den Sperrmodus an, der für eine Informix-Datenquelle festgelegt werden soll. Der Informix-Wrapper setzt den Befehl 'SET LOCK MODE' sofort ab, nachdem die Verbindung zu einer Informix-Datenquelle hergestellt wurde. Gültige Werte sind:</p> <p><b>'W'</b> Setzt den Informix-Sperrmodus auf WAIT. Wenn der Wrapper versucht, auf eine gesperrte Tabelle oder Zeile zuzugreifen, wartet Informix, bis die Sperre aufgehoben wird.</p> <p><b>'N'</b> Setzt den Informix-Sperrmodus auf NOWAIT. Wenn der Wrapper versucht, auf eine gesperrte Tabelle oder Zeile zuzugreifen, gibt Informix einen Fehler zurück.</p> <p><b>'n'</b> Setzt den Informix-Sperrmodus für <i>n</i> Sekunden auf WAIT. Wenn der Wrapper versucht, auf eine gesperrte Tabelle oder Zeile zuzugreifen, und die Sperre wird nicht innerhalb der angegebenen Anzahl Sekunden aufgehoben, gibt Informix einen Fehler zurück.</p>	'W'

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
IO_RATIO	<p>Gibt den Geschwindigkeitsunterschied zwischen dem Ein-/Ausgabesystem einer Datenquelle und dem Ein-/Ausgabesystem des Servers mit zusammengeschlossenen Datenbanken an.</p> <p>Gültige Werte liegen im Bereich zwischen größer als 0 und <math>1 \times 10^{23}</math>. Die Werte können in jeder zulässigen REAL-Notation definiert werden.</p> <p>Die Einstellung 1 gibt an, dass das Ein-/Ausgabesystem des DB2-Servers mit zusammengeschlossenen Datenbanken und das Ein-/Ausgabesystem der Datenquelle die gleiche Geschwindigkeit haben, also ein Verhältnis von 1:1. Die Einstellung .5 gibt an, dass die Geschwindigkeit des Ein-/Ausgabesystems des DB2-Servers mit zusammengeschlossenen Datenbanken um 50% unter der Geschwindigkeit des Ein-/Ausgabesystems der Datenquelle liegt. Die Einstellung 2 gibt an, dass die Geschwindigkeit des Ein-/Ausgabesystems des DB2-Servers mit zusammengeschlossenen Datenbanken doppelt so hoch ist wie die Geschwindigkeit des Ein-/Ausgabesystems der Datenquelle.</p>	'1.0'
IUD_APP_SVPT_ENFORCE	<p>Gibt an, ob das DB2-System zusammengeschlossener Datenbanken das Finden oder Erstellen von Anweisungen für Anwendungssicherungspunkte durchsetzen soll. Wenn diese Option mit der Anweisung SET SERVER OPTION gesetzt wird, hat diese Option keine Auswirkung bei statischen SQL-Anweisungen.</p> <p>'Y' Wenn in INSERT-, UPDATE- und DELETE-Transaktionen ein Fehler auftritt und für die Datenquelle Anweisungen für Anwendungssicherungspunkte nicht zwingend verwendet werden, macht der Server mit zusammengeschlossenen Datenbanken diese fehlerhaften Transaktionen mit einer ROLLBACK-Operation rückgängig. In diesem Fall wird der SQL-Fehlercode SQL1476N zurückgegeben.</p> <p>'N' Der Server mit zusammengeschlossenen Datenbanken macht beim Auftreten eines Fehlers Transaktionen nicht rückgängig. Die Fehlerbehebung muss von Ihrer Anwendung übernommen werden.</p>	'Y'
LOGIN_TIMEOUT	Gibt die Anzahl der Sekunden an, die der DB2-Server mit zusammengeschlossenen Datenbanken auf eine Antwort von Sybase Open Client auf die Anmeldeanforderung warten muss. Die Standardwerte entsprechen den Standardwerten für TIMEOUT.	'0'



Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
MAX_ROWS	<p>Gibt die Anzahl der Zeilen an, die der Server mit zusammengeschlossenen Datenbanken für eine Abfrage zurückgibt, die den Entrez-Wrapper verwendet.</p> <p>Sie können nur positive Zahlen und Null angeben. Wenn Sie für die Option den Wert Null definieren, können Abfragen eine unbegrenzte Anzahl von Zeilen von der NCBI-Website abrufen. Wird hingegen die Serveroption MAX_ROWS auf den Wert Null oder einen sehr hohen Wert gesetzt, kann sich dies auf die Abfrageleistung des Systems auswirken.</p> <p>Die Serveroption MAX_ROWS ist nicht erforderlich.</p>	<p>Microsoft Windows-Betriebssysteme: 2000 Zeilen.</p> <p>UNIX-Betriebssysteme: 5000 Zeilen.</p>
MQ_CONN_NAME	<p>Der Hostname oder die Netzadresse des Computers, auf dem der WebSphere MQ-Server ausgeführt wird. Beispiel für einen Verbindungsnamen: 9.30.76.151(1420), wobei 1420 die Portnummer darstellt. Wenn die Portnummer ausgeschlossen ist, wird der Standardwert 1414 verwendet. Diese Option ist optional. Wird sie nicht angegeben, wird die Umgebungsvariable MQSERVER zur Auswahl der Kanaldefinition benutzt (sofern diese in der Datei db2dj.ini definiert wurde). Wenn MQSERVER nicht definiert ist, wird die Clientkanaltabelle verwendet.</p>	<p>Der Wrapper verwendet die Umgebungsvariable MQSERVER, sofern diese in der Datei db2dj.ini angegeben wurde, zur Auswahl der Kanaldefinition. Wenn die Umgebungsvariable MQSERVER nicht definiert wurde, verwendet der Wrapper die Clientkanaltabelle.</p>
MQ_MANAGER	<p>Der Name des WebSphere MQ-Managers. Jeder gültige Name für den WebSphere MQ-Manager. Diese Option ist erforderlich.</p>	keine
MQ_RESPONSE_TIMEOUT	<p>Die Zeitdauer, die der Wrapper auf eine antwortabhängige Nachricht aus der Antwortwarteschlange wartet. Der Wert wird in Millisekunden angegeben. Mit dem Sonderwert -1 können Sie angeben, dass kein Zeitlimitintervall festgelegt werden soll. Diese Option ist optional.</p>	10000
MQ_SVRCONN_CHANNELNAME	<p>Der Name des Serververbindungskanals im WebSphere MQ-Manager, zu dem der Wrapper eine Verbindung aufbauen soll. Dieser Parameter kann nur angegeben werden, wenn die Serveroption MQ_CONN_NAME angegeben wurde. Der Standardkanal für die Serververbindung lautet SYSTEM.DEF.SVRCONN und wird verwendet, wenn diese Option nicht angegeben wird.</p>	SYSTEM.DEF.SVRCONN

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
NODE	<p>Relationale Datenquellen: Der Name, mit dem eine Datenquelle im zugehörigen Verwaltungssystem für relationale Datenbanken (RDBMS) als Exemplar definiert wird.</p> <p>Documentum: Gibt den tatsächlichen Namen für die Documentum-Doabase an. Diese Option ist erforderlich.</p> <p>BLAST: Gibt den Hostnamen des Systems an, auf dem der BLAST-Dämonprozess ausgeführt wird. Diese Option ist erforderlich.</p> <p>HMMER: Gibt den Hostnamen des Servers an, auf dem der HMMER-Dämonprozess ausgeführt wird. Diese Option ist erforderlich.</p> <p>BioRS: Gibt den Hostnamen des Systems an, auf dem das BioRS-Abfragetool bereitgestellt wird. Diese Option ist optional.</p>	BioRS: <i>localhost</i>
OS_TYPE	Gibt das Betriebssystem des Doabase-Servers an. Gültige Werte sind AIX, SOLARIS und WINDOWS. Diese Option ist erforderlich.	keine
PACKET_SIZE	Gibt die Paketgröße der Schnittstellendatei für Sybase in Byte an. Unterstützt die Datenquelle die angegebene Paketgröße nicht, schlägt die Verbindung fehl. Sind die einzelnen Datensätze sehr groß (beispielsweise durch das Einfügen von Zeilen in große Tabellen), kann die Leistung durch Vergrößern der Paketgröße erheblich verbessert werden. Die Bytgröße ist ein numerischer Wert.	
PASSWORD	<p>Gibt an, ob Kennwörter an eine Datenquelle gesendet werden.</p> <p>'Y' Kennwörter werden an die Datenquelle gesendet und geprüft.</p> <p>'N' Kennwörter werden nicht an die Datenquelle gesendet und geprüft.</p>	'Y'

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
PLAN_HINTS	<p>Gibt an, ob <i>Planhinweise</i> aktiviert werden sollen. Planhinweise sind Anweisungsfragmente, die Optimierungsprogrammen von Datenquellen zusätzliche Informationen zur Verfügung stellen. Bei bestimmten Abfragetypen kann mit Hilfe dieser Informationen die Abfrageleistung verbessert werden. Die Planhinweise können das Optimierungsprogramm der Datenquelle bei der Entscheidung unterstützen, ob und welcher Index oder welche Tabellenverknüpfungssortierung verwendet werden soll.</p> <p>'Y' In der Datenquelle sollen Planhinweise aktiviert werden, falls diese von der Datenquelle unterstützt werden.</p> <p>'N' In der Datenquelle sollen keine Planhinweise aktiviert werden.</p> <p>Diese Option steht nur für Oracle- und Sybase-Datenquellen zur Verfügung.</p>	'N'
PORT	Gibt die Nummer des Ports an, der vom Wrapper zur Herstellung der Verbindung zum BioRS-Server verwendet wird. Diese Option ist optional.	'5014'
PROCESSORS	Gibt die Anzahl der Prozessoren an, die das Programm HMMER verwendet. Diese Option entspricht der Option <code>--cpu</code> des Befehls <code>hmmpfam</code> .	keine
PROXY_AUTHID	Gibt den Benutzernamen an, der verwendet werden soll, wenn für PROXY_TYPE der Wert 'SOCKS5' definiert ist. Dieses Feld ist optional, wenn für PROXY_TYPE der Wert 'SOCKS5' verwendet wird. Wenden Sie sich an den zuständigen Netzadministrator, wenn Sie Informationen zu dem zu verwendenden Benutzernamen benötigen. Diese Option ist ungültig, wenn für PROXY_TYPE nicht 'SOCKS5' angegeben ist.	keine
PROXY_PASSWORD	Gibt das Kennwort an, das verwendet werden soll, wenn für PROXY_TYPE der Wert 'SOCKS5' verwendet wird. Dieses Feld ist optional, wenn für PROXY_TYPE der Wert 'SOCKS5' verwendet wird. Wenden Sie sich an den zuständigen Netzadministrator, wenn Sie Informationen zu dem zu verwendenden Kennwort benötigen. Diese Option ist ungültig, wenn für PROXY_TYPE nicht 'SOCKS5' angegeben ist.	keine
PROXY_SERVER_NAME	Gibt den Proxy-Server-Namen oder die IP-Adresse an. Dieses Feld ist erforderlich, wenn für PROXY_TYPE der Wert 'HTTP', 'SOCKS4' oder 'SOCKS5' definiert ist. Wenden Sie sich an den zuständigen Netzadministrator, wenn Sie Informationen zum Proxy-Server-Namen oder zur IP-Adresse benötigen.	keine

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
PROXY_SERVER_PORT	Gibt die Portnummer des Proxy-Servers an. Dieses Feld ist erforderlich, wenn für PROXY_TYPE der Wert 'HTTP', 'SOCKS4' oder 'SOCKS5' definiert ist. Wenden Sie sich an den zuständigen Netzadministrator, wenn Sie Informationen zu der zu verwendenden Portnummer des Proxy-Servers benötigen.	keine
PROXY_TYPE	Gibt den Proxytyp an, der für den Internetzugriff verwendet wird, wenn sich Ihr System hinter einer Firewall befindet. Die gültigen Werte lauten 'NONE', 'HTTP', 'SOCKS4' und 'SOCKS5'. Der Standardwert ist 'NONE'. Wenden Sie sich an den zuständigen Netzadministrator, wenn Sie Informationen zum verwendeten Proxytyp benötigen.	'NONE'
PUSHDOWN	<p>'Y' DB2 UDB überprüft die Möglichkeit, die Auswertung von Operationen durch die Datenquelle durchführen zu lassen.</p> <p>'N' DB2 UDB sendet nur diejenigen SQL-Anweisungen an die Datenquelle, die für die Spaltennamen ausschließlich SELECT-Anweisungen enthalten. Vergleichselemente (wie z. B. WHERE=), Spalten- und Skalarfunktionen (wie MAX und MIN), Sortierungen (wie ORDER BY oder GROUP BY) und Verknüpfungen sind in den an die Datenquelle gesendeten SQL-Anweisungen nicht enthalten.</p>	'Y'
RDBMS_TYPE	Gibt das Verwaltungssystem für relationale Datenbanken (RDBMS) an, das von der Doabase verwendet wird. Gültige Werte sind DB2, INFORMIX, ORACLE, SQLSERVER und SYBASE. Diese Option ist erforderlich.	keine
RESPONSE_QUEUE	Der Name der Antwortwarteschlange, die Abfrageergebnisse vom Adapter zum Wrapper weiterleitet. Der Name muss mit den Spezifikationen für die Angabe von Warteschlangennamen in WebSphere MQ übereinstimmen. Diese Option ist erforderlich.	keine
REQUEST_QUEUE	Der Name der Anforderungwarteschlange, die Abfrageanforderungen vom Wrapper zum Adapter weiterleitet. Der Name muss mit den Spezifikationen für die Angabe von Warteschlangennamen in WebSphere MQ übereinstimmen. Diese Option ist erforderlich.	keine
SOCKET_TIMEOUT	Gibt die maximal zulässige Zeit in Minuten an, die der DB2-Server mit zusammengeschlossenen Datenbanken auf die Ergebnisse vom Proxy-Server wartet. Zulässige Werte sind alle Zahlen, die größer-gleich Null sind. Der Standardwert ist Null ('0'). Der Wert Null gibt an, dass unbegrenzt gewartet wird.	0

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
TIMEFORMAT	Das Zeitformat, das von der Datenquelle verwendet wird. Geben Sie das Format unter Verwendung von 'hh24', 'mm' oder 'ss' ein. Zur Darstellung des Zeitformats für '16:00:00' beispielsweise verwenden Sie 'hh24:mm:ss'. Zur Darstellung des Zeitformats für '08:00:00' verwenden Sie 'hh24:mm:ss'. Für dieses Feld ist die Dateneingabe optional.	keine
TIMESTAMPFORMAT	Das Zeitmarkenformat, das von der Datenquelle verwendet wird. Das Format entspricht dem Format für Datum und Zeit und enthält zusätzlich 'n' für eine Zehntelsekunde, 'nn' für eine Hundertstelsekunde, 'nnn' für Millisekunden und so weiter bis zu 'nnnnnn' für Mikrosekunden. Zur Darstellung des Zeitmarkenformats für '2003-01-01-24:00:00.000000' beispielsweise verwenden Sie 'JJJJ-MM-TT-hh24:mm:ss.nnnnnn'. Für dieses Feld ist die Dateneingabe optional.	keine
TIMEOUT	<p>Sybase: Gibt die Anzahl der Sekunden an, die der DB2-Server mit zusammengeschlossenen Datenbanken auf eine Sybase Open Client-Antwort für eine beliebige SQL-Anforderung wartet. Der Wert für <i>Sekunden</i> ist eine positive ganze Zahl im Integerbereich von DB2 Universal Database. Der in der Option TIMEOUT für das Zeitlimit angegebene Wert ist vom verwendeten Wrapper abhängig. Der Standardwert der Option TIMEOUT für Sybase-Wrapper ist 0. Dieser Wert bewirkt, dass DB2 UDB unbegrenzt auf eine Antwort wartet.</p> <p>BioRS: Gibt die Zeit (in Minuten) an, die der BioRS-Wrapper auf eine Antwort vom BioRS-Server wartet. Der Standardwert ist 10. Diese Option ist optional.</p>	'0'; BioRS: '10'
TRANSACTIONS	<p>Gibt den Transaktionsmodus des Servers an. Gültige Werte sind:</p> <p>'NONE' Es sind keine Transaktionen aktiviert.</p> <p>'QUERY' Die Transaktionen sind nur für Dctm_Query-Methoden aktiviert.</p> <p>'ALL' Transaktionen sind für die Dctm_Query-Methode aktiviert. Im aktuellen Release weist ALL dieselbe Funktion auf wie QUERY.</p>	'QUERY'
USE_CLOB_SEQUENCE	Diese Option gibt den Datentyp an, die der Server mit zusammengeschlossenen Datenbanken für die Spalte BlastSeq oder HmmQSeq verwendet. Der Wert kann 'Y' oder 'N' lauten. Sie können die Anweisung CREATE NICKNAME oder ALTER NICKNAME verwenden, um den Standarddatentyp für die Spalte BlastSeq oder HmmQSeq zu überschreiben.	'Y'

Tabelle 148. Serveroptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
VARCHAR_NO_TRAILING_BLANKS	<p>Diese Option gilt für Datenquellen mit variablen Zeichendatentypen, die fehlende Zeichen bis zur maximal zulässigen Länge bei Vergleichsoperationen nicht mit folgenden Leerzeichen auffüllen.</p> <p>Einige Datenquellen, wie z. B. Oracle, verfügen nicht über eine Vergleichssemantik mit Leerzeichen als Füllzeichen, die dieselben Ergebnisse zurückgibt wie die Vergleichssemantik von DB2 für Linux, UNIX und Windows. Aktivieren Sie diese Option, wenn sie für alle VARCHAR- oder VARCHAR2-Spalten in Datenquellenobjekten, auf die der angegebene Server zugreift, angewendet werden soll. Sichten sind mit eingeschlossen.</p> <p>Y Diese VARCHAR-Spalten enthalten keine abschließenden Leerzeichen, oder diese Datenquelle verfügt über eine Vergleichssemantik mit Leerzeichen als Füllzeichen, die mit der Vergleichssemantik des Servers mit zusammengeschlossenen Datenbanken übereinstimmt.</p> <p>Der Server mit zusammengeschlossenen Datenbanken sendet Zeichenvergleichsoperationen im Pushdown-Modus zur Verarbeitung an die Datenquelle.</p> <p>N Diese VARCHAR-Spalten enthalten folgende Leerzeichen, und die Datenquelle verfügt über eine Vergleichssemantik mit Leerzeichen als Füllzeichen, die von der Vergleichssemantik des Servers mit zusammengeschlossenen Datenbanken abweicht.</p> <p>Der Server mit zusammengeschlossenen Datenbanken verarbeitet Zeichenvergleichsoperationen, wenn ein Ausgleich für äquivalente Semantiktypen nicht möglich ist. Dies gilt z. B. beim erneuten Schreiben des Vergleichselements.</p>	N für betroffene Datenquellen.

Anmerkungen zu dieser Tabelle:

1. Diese Feld findet unabhängig von dem für die Authentifizierung angegebenen Wert Anwendung.
2. Da DB2 UDB Benutzer-IDs in Großschreibung speichert, sind die Werte 'N' und 'U' logisch äquivalent.
3. Die Einstellung für FOLD\_PW ist unwirksam, wenn die Einstellung für das Kennwort 'N' lautet. Da kein Kennwort gesendet wird, ist die Schreibweise unerheblich.

- |
- | 4. Vermeiden Sie Nullwerte für diese Optionen. Die Angabe eines Nullwertes
- | kann auf den ersten Blick sinnvoll erscheinen, da DB2 UDB dann mehrere Ver-
- | suche unternimmt, Benutzer-IDs und Kennwörter aufzulösen. Diese Vorgehens-
- | weise kann jedoch die Leistung des Systems beeinträchtigen. (In bestimmten
- | Fällen ist es möglich, dass DB2 UDB eine Benutzer-ID und das zugehörige
- | Kennwort viermal senden muss, bevor die Authentifizierung durch die Daten-
- | quelle erfolgreich ausgeführt werden kann.)

|

| **Zugehörige Konzepte:**

- |
- „Serverkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ in *Systeme zusammengesetzter Datenbanken*
  - „Serverkenndaten mit Auswirkungen auf die globale Optimierung“ in *Systeme zusammengesetzter Datenbanken*
- |

|

| **Zugehörige Tasks:**

- |
- „Registrieren von Serverdefinitionen für eine Datenquelle“ auf Seite 71

|

| **Zugehörige Referenzen:**

- |
- „DROP statement“ in *SQL Reference, Volume 2*
  - „ALTER SERVER statement“ in *SQL Reference, Volume 2*
  - „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- |

## Anhang E. Benutzerzuordnungsoptionen für Systeme zusammengesessener Datenbanken

Diese Optionen sind für alle relationalen Datenquellen zulässig. Bei nicht relationalen Datenquellen sind die Optionen REMOTE\_AUTHID und REMOTE\_PASSWORD bei den folgenden Datenquellen zulässig: BioRS, Documentum, Extended Search und Web-Services. Die Option GUEST ist für BioRS-Datenquellen zulässig.

Diese Optionen werden in den Anweisungen CREATE USER MAPPING und ALTER USER MAPPING verwendet.

Tabelle 149. Benutzerzuordnungsoptionen und deren Einstellungen

Option	Gültige Einstellungen	Standardeinstellung
ACCOUNTING	DRDA: Wird zur Angabe einer DRDA-Abrechnungszeichenfolge verwendet. Gültige Einstellungen sind alle Zeichenfolgen mit einer Länge von 255 oder weniger. Diese Option ist nur dann erforderlich, wenn Abrechnungsinformationen weitergegeben werden müssen. Weitergehende Informationen enthält das Benutzerhandbuch zu DB2 Connect.	keine
GUEST	Gibt an, ob der Wrapper für den BioRS-Server den Zugriffsmodus für Gastmaschinen verwenden soll.  Y Der Wrapper verwendet für den BioRS-Server den Zugriffsmodus für Gastmaschinen.  N Der Wrapper verwendet für den BioRS-Server nicht den Zugriffsmodus für Gastmaschinen.  Wird für diese Option der Wert Y definiert, schließt sich die Verwendung dieser Option und der Optionen REMOTE_AUTHID und REMOTE_PASSWORD gegenseitig aus.	N
REMOTE_AUTHID	Gibt die in der Datenquelle verwendete Berechtigungs-ID an. Gültige Einstellungen sind alle Zeichenfolgen mit einer Länge von 255 oder weniger.	Die Berechtigungs-ID, die für die Herstellung der Verbindung zu DB2 Universal Database verwendet wird.
REMOTE_DOMAIN	Documentum: Gibt die Windows NT-Domäne an, die für die Authentifizierung von Benutzern verwendet wird, die eine Verbindung zu einer Documentum-Datenquelle herstellen wollen. Gültige Einstellungen schließen jeden beliebigen Windows NT-Domänennamen ein.	Die Standarddomäne zur Authentifizierung für die Documentum-Datenbank.



Tabelle 149. Benutzerzuordnungsoptionen und deren Einstellungen (Forts.)

Option	Gültige Einstellungen	Standardeinstellung
REMOTE_PASSWORD	<p>Gibt das in der Datenquelle verwendete Berechtigungs-kennwort an. Gültige Einstellungen sind alle Zeichen-folgen mit einer Länge von 32 oder weniger.</p> <p>Diese Option muss nicht definiert werden, wenn die folgenden Bedingungen gelten:</p> <ul style="list-style-type: none"> <li>• Der Konfigurationsparameter AUTHENTICATON des Datenbankmanagers ist auf SERVER gesetzt.</li> <li>• Bei der Herstellung der Verbindung zur DB2-Daten-bank haben Sie eine Berechtigungs-ID und ein zuge-höriges Kennwort angegeben.</li> </ul> <p>Wenn auf Ihrem Server ein Kennwort erforderlich ist, und Sie diese Option nicht definiert haben, müssen Sie sicherstellen, dass die beiden zuvor aufgeführten Bedin-gungen erfüllt sind. Andernfalls schlägt der Verbindungsaufbau fehl.</p>	Das Kennwort, das Sie zur Herstellung der Verbindung zu DB2 Universal Database verwenden, wenn beide in den Spalten für zulässige Ein-stellungen aufgeführten Bedingungen erfüllt sind.

**Zugehörige Konzepte:**

- „DB2 Connect und DRDA“ in *DB2 Connect Benutzerhandbuch*
- „DRDA und Datenzugriff“ in *DB2 Connect Benutzerhandbuch*

**Zugehörige Tasks:**

- „Registrieren von Benutzerzuordnungen für eine Datenquelle“ auf Seite 72

## Anhang F. Kurznamenoptionen für Systeme zusammengesetzter Datenbanken

Tabelle 150 und Tabelle 151 enthalten eine Liste der Kurznamenoptionen für die verschiedenen Datenquellen. Tabelle 152 auf Seite 662 beschreibt die verschiedenen Kurznamenoptionen und listet die zulässigen sowie die Standardeinstellungen auf.

Tabelle 150. Verfügbare Kurznamenoptionen - A bis P

Datenquelle	ALL_VERSIONS	APPLICATIONID	BUSOBJ_NAME	CATEGORY	COLUMN_DELIMITER	DATASOURCE	DIRECTORY_PATH	FILE_PATH	FOLDERS	HMMTYPE	INSTANCE_PARSE_TIME	IS_REG_TABLE	KEY_COLUMN	MAXHIT	NAMESPACES	NEXT_TIME	PARENT	PROCESSORS
BioRS																		
BLAST						X												X
Documentum	X								X			X						
Entrez																	X	
Excel								X										
Extended Search		X		X		X							X					
HMMER						X				X								
Dateien mit Tabellenstruktur					X			X					X					
Web-Services															X			
WebSphere Business Integration			X												X			
XML							X	X			X					X		

Tabelle 151 enthält eine Auflistung der Kurznamenoptionen (R bis X) der einzelnen Datenquellen.

Tabelle 151. Verfügbare Kurznamenoptionen - R bis X

Datenquelle	RANGE	REMOTE_OBJECT	SOAPACTION	SORTED	SORTFIELD	SORTORDER	STREAMING	TEMPLATE	TOTALMAXHIT	TIMEOUT	URL	VALIDATE	VALIDATE_DATA_FILE	VERTICAL_TABLE	XPATH	XPATH_EVAL_TIME
BioRS		X														
BLAST										X						
Documentum		X														

Tabelle 151. Verfügbare Kurznamenoptionen - R bis X (Forts.)

Datenquelle	RANGE	REMOTE_OBJECT	SOAPACTION	SORTED	SORTFIELD	SORTORDER	STREAMING	TEMPLATE	TOTALMAXHIT	TIMEOUT	URL	VALIDATE	VALIDATE_DATA_FILE	VERTICAL_TABLE	XPATH	XPATH_EVAL_TIME
Entrez		X														
Excel	X															
Extended Search					X	X			X	X				X		
HMMER										X						
Dateien mit Tabellenstruktur				X									X			
Web-Services			X				X	X			X				X	
WebSphere Business Integration								X							X	
XML							X					X			X	X

Tabelle 152 beschreibt die verschiedenen Kurznamenoptionen und listet die zulässigen sowie die Standardeinstellungen auf.

Tabelle 152. Kurznamenoptionen und zugehörige Einstellungen

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
ALL_VERSIONS	Gibt an, ob alle Objektversionen gesucht werden sollen. Die zulässigen Werte sind y, Y, n und N. Der Standardwert N gibt an, dass nur die aktuellen Objektversionen in die Abfrageverarbeitung eingebunden werden. Diese Option ist nicht zulässig, wenn für IS_REG_TABLE = 'Y' definiert wurde.	N
APPLICATIONID	Gibt den Namen der Extended Search-Anwendung an, die gesucht werden soll. Dieser Name muss in der Extended Search-Konfigurationsdatenbank vorhanden sein. Diese Option ist erforderlich.	
BUSOBJ_NAME	Der Name der XML-Schemadefinitionsdatei (.xsd), die für das Geschäftsobjekt steht. Beispiel: sap_bapi_customer_get_detail2. Diese Option muss in einem übergeordneten Kurznamen angegeben werden.	
CATEGORY	Gibt eine oder mehrere Extended Search-Kategorien an, die gesucht werden sollen. Wenn Sie diese Option nicht benutzen, müssen Sie mindestens einen Datenquellennamen angeben. Zur Angabe mehrerer Kategorien müssen Sie die einzelnen Kategoriennamen durch ein Semikolon trennen.	

Tabelle 152. Kurznamenoptionen und zugehörige Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
COLUMN_DELIMITER	<p>Der zum Trennen von Spalten einer Datei mit Tabellenstruktur verwendete Begrenzer in einfachen Anführungszeichen. Der Spaltenbegrenzer kann mehrere Zeichen umfassen. Wenn kein Spaltenbegrenzer definiert wurde, wird standardmäßig ein Komma verwendet. Ein einfaches Anführungszeichen kann nicht als Begrenzer eingesetzt werden. Der Spaltenbegrenzer muss innerhalb einer Datei einheitlich verwendet werden. Ein Nullwert wird durch zwei neben einander stehende Begrenzer oder einen Begrenzer mit nachfolgendem Zeilenabschlusszeichen dargestellt, wenn das Feld NULL innerhalb der Zeile an letzter Stelle steht. Die Spaltenbegrenzer können nicht als reguläre Spaltendaten verwendet werden.</p>	<p>Als Standardbegrenzer wird ein Komma verwendet.</p>
DATASOURCE	<p>Für Extended Search: Gibt eine oder mehrere Extended Search-Datenquellen an, die durchsucht werden sollen. Wenn Sie diese Option nicht benutzen, müssen Sie mindestens einen Kategoriennamen angeben. Zur Angabe mehrerer Datenquellen müssen die Datenquellennamen mit einem Semikolon voneinander getrennt werden.</p> <p>Für BLAST: Der Name der Datenquelle, in der die BLAST-Suche ausgeführt werden soll. Die hier verwendete Zeichenfolge muss auch in der Konfigurationsdatei des BLAST-Dämons definiert sein. Diese Option ist erforderlich.</p> <p>Für HMMER (Typ PFAM): Der Name der HMM Profile-Datenbank, die mit HMMPFAM durchsucht werden soll. Die hier verwendete Zeichenfolge muss auch in der Konfigurationsdatei des HMMER-Dämons definiert sein. Diese Option ist erforderlich.</p> <p>Für HMMER (Typ SEARCH): Der Name der Sortierdatei, die mit HMMSEARCH durchsucht werden soll. Die hier verwendete Zeichenfolge muss auch in der Konfigurationsdatei des HMMER-Dämons definiert sein. Diese Option ist erforderlich.</p>	

Tabelle 152. Kurznamenoptionen und zugehörige Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
DIRECTORY_PATH	<p>Gibt den Pfadnamen eines Verzeichnisses an, das eine oder mehrere XML-Dateien enthält. Verwenden Sie diese Option, um für mehrere XML-Quellendateien einen einzigen Kurznamen zu erstellen. Der XML-Wrapper verwendet nur Dateien mit der Erweiterung .xml, die im angegebenen Verzeichnis gespeichert sind. Alle anderen Dateien in diesem Verzeichnis werden vom XML-Wrapper hingegen ignoriert. Wenn Sie diese Kurznamenoption angeben, dann darf keine DOCUMENT-Spalte angegeben werden. Diese Option ist nur für den Rootkurznamen (d. h. den Kurznamen für die Elemente der obersten Hierarchieebene des XML-Dokuments) zulässig.</p>	
FILE_PATH	<p>Für Microsoft Excel: Gibt den vollständig qualifizierten Verzeichnispfad und Dateinamen des Excel-Arbeitsblattes an, auf das Sie zugreifen wollen. Diese Option ist erforderlich.</p> <p>Für Dateien mit Tabellenstruktur: Der vollständig qualifizierte Pfad zur Datei mit Tabellenstruktur, auf die zugegriffen werden soll, in einfachen Anführungszeichen. An Stelle einer Pipe oder eines anderen nicht dem Standard entsprechenden Dateityps muss eine Standarddatei oder eine symbolische Verbindung verwendet werden. Sie müssen entweder die Kurznamenoption FILE_PATH oder DOCUMENT angeben. Wenn die Kurznamenoption FILE_PATH angegeben wird, darf die Kurznamenspaltenoption DOCUMENT nicht angegeben werden.</p> <p>Für XML: Gibt den Dateipfad des XML-Dokuments an. Wenn Sie diese Kurznamenoption angeben, dann darf keine DOCUMENT-Spalte angegeben werden. Diese Option ist nur für den Rootkurznamen (d. h. den Kurznamen für die Elemente der obersten Hierarchieebene des XML-Dokuments) zulässig.</p>	
FOLDERS	<p>Gibt eine Zeichenfolge an, die ein oder mehrere logisch kombinierte und syntaktisch korrekte Documentum-FOLDER-Vergleichselemente enthält. Die Angabe von FOLDER-Vergleichselementen schränkt die Gruppe der Dokumente, die durch diesen Kurznamen dargestellt werden, auf die Dokumente in den dedizierten Ordnern ein.</p> <p>Wenn Sie diese Option angeben, schließen Sie den gesamten Wert der Option FOLDERS in einfache Anführungszeichen ein, und verwenden Sie innerhalb der Zeichenfolge an Stelle einfacher Anführungszeichen doppelte Anführungszeichen.</p> <p>Diese Option ist nicht zulässig, wenn für IS_REG_TABLE = 'Y' definiert wurde.</p>	

Tabelle 152. Kurznamenoptionen und zugehörige Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
HMMTYPE	Optional: Das Alphabet, das sowohl in Modellen als auch Gensequenzen benutzt wird. Der Wert kann entweder NUCLEIC oder PROTEIN lauten, wobei die Groß-/Kleinschreibung nicht beachtet werden muss.	PROTEIN
INSTANCE_PARSE_TIME	Gibt die Zeit (in Millisekunden) an, die zur syntaktischen Analyse der Daten einer Zeile im XML-Quellendokument benötigt wird. Sie können die Optionen INSTANCE_PARSE_TIME, XPATH_EVAL_TIME und NEXT_TIME ändern, um Abfragen in umfangreichen oder komplexen XML-Quellenstrukturen zu optimieren. Diese Option ist nur für Spalten des Rootkurznamens (d. h. den Kurznamen für die Elemente der obersten Hierarchieebene des XML-Dokuments) zulässig. Die angegebene Zahl kann ein Integer- oder ein Dezimalwert sein.	7
IS_REG_TABLE	Gibt an, ob es sich bei dem in der Option REMOTE_OBJECT angegebenen Objekt um eine registrierte Documentum-Tabelle handelt. Die gültigen Werte sind 'y', 'Y', 'n' und 'N'.  Sie können keinen Kurznamen von einem Documentum-Objekt in eine registrierte Tabelle (oder zurück) ändern, indem Sie diese Option mit der Anweisung ALTER NICKNAME ändern. Stattdessen müssen Sie den Kurznamen löschen und anschließend erneut erstellen.	N
KEY_COLUMN	Der Name der Spalte in der Datei, die als Schlüssel für die Sortierung des Dateiinhalts verwendet wird, in einfachen Anführungszeichen. Verwenden Sie diese Option nur für sortierte Dateien. Eine Spalte, die mit der Kurznamenspaltenoption DOCUMENT dediziert wurde, darf nicht als Schlüsselspalte angegeben werden.  Es werden nur einspaltige Schlüssel unterstützt. Die Verwendung von mehrspaltigen Schlüsseln ist nicht zulässig. Als Wert muss der Name einer Spalte angegeben werden, die in der Anweisung CREATE NICKNAME definiert wurde. Die Spalte muss in aufsteigender Reihenfolge sortiert sein. In der Schlüsselspalte dürfen keine Nullwerte zulässig sein. Fügen Sie hierzu in der Kurznamenanweisung die Option NOT NULL zur Spaltendefinition hinzu.  Bei dieser Option muss die Groß-/Kleinschreibung beachtet werden. In DB2 UDB werden Spaltennamen jedoch in Großbuchstaben umgesetzt, wenn der Spaltenname nicht in doppelten Anführungszeichen steht.	Wenn der Wert für einen sortierten Kurznamen nicht angegeben ist, stimmt dieser mit dem Namen der ersten Spalte in der Datei mit Kurznamen überein.
MAXHIT	Ein INTEGER-Wert, der die maximal zulässige Anzahl von Ergebnissen angibt, die aus durchsuchten Quellen zurückgegeben werden kann.	50

Tabelle 152. Kurznamenoptionen und zugehörige Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
NAMESPACES	<p>Die Namensbereiche, die den Namensbereichspräfixen zugeordnet sind, die in den Optionen XPATH und TEMPLATE der Spalten benutzt werden. Die Syntax lautet wie folgt:</p> <pre>NAMESPACES 'prefix1="actual_namespace1", prefix2="actual_namespace2" '</pre> <p>Trennen Sie die einzelnen Namensbereiche durch ein Komma. Beispiel:</p> <pre>NAMESPACES ' c="http://www.myweb.com/cust", i="http://www.myweb.com/cust/id", n="http://www.myweb.com/cust/name" '</pre>	
NEXT_TIME	<p>Gibt die Uhrzeit (in Millisekunden) an, die zum Auffinden nachfolgender Quellenelemente in einem XPath-Ausdruck erforderlich ist. Sie können die Optionen NEXT_TIME, XPATH_EVAL_TIME und INSTANCE_PARSE_TIME ändern, um Abfragen in umfangreichen oder komplexen XML-Quellenstrukturen zu optimieren. Diese Option ist für Rootkurznamen und Nicht-Rootkurznamen zulässig.</p>	1
PARENT	<p>Diese Option wird nur für untergeordnete Kurznamen angegeben, deren übergeordnetes Element mit Hilfe der Option REMOTE_OBJECT umbenannt wurde. Die Option PARENT stellt eine Zuordnung zwischen einem unter- und einem übergeordneten Element her, wenn in einem DB2-Schema mehrere Kurznamengruppen definiert sind. Bei diesem Namen muss die Groß-/Kleinschreibung beachtet werden.</p>	
PROCESSORS	<p>Gibt die Anzahl der Prozessoren an, die bei der Auswertung einer BLAST-Abfrage verwendet werden sollen. Diese Option entspricht der Option blastall -a.</p>	1
RANGE	<p>Gibt einen Bereich von Zellen an, die in der Datenquelle verwendet werden sollen.</p>	

Tabelle 152. Kurznamenoptionen und zugehörige Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
REMOTE_OBJECT	<p>Für BioRS: Gibt den Namen der BioRS-Datenbank an, die dem Kurznamen zugeordnet ist. Dieser Name legt das Schema und die BioRS-Datenbank für den Kurznamen fest. Dieser Name gibt auch die Beziehung des Kurznamens zu anderen Kurznamen an. Die Regelungen für die Groß-/Kleinschreibung dieses Namens hängen von den entsprechenden Regelungen für den BioRS-Server und vom Wert für die Serveroption CASE_SENSITIVE ab. Die Anweisung ALTER NICKNAME kann nicht zum Ändern oder Löschen dieses Namens verwendet werden. Wenn der Name der in dieser Option verwendeten BioRS-Datenbank sich ändert, müssen Sie den Kurznamen löschen und anschließend vollständig neu erstellen.</p> <p>Für Documentum: Gibt den Namen des Documentum-Objektyps an, der dem Kurznamen zugeordnet ist. Als Name kann jeder Documentum-Objektyp bzw. jede registrierte Tabelle verwendet werden. Dem Namen einer registrierten Tabelle muss der Name des Tabelleneigners als Präfix vorangestellt werden. Wenn die registrierte Tabelle dem Doctype-Eigner zugeordnet ist, kann der Wert dm_dbo für den Eignernamen benutzt werden. Diese Option ist erforderlich. Wenn Sie die Anweisung ALTER NICKNAME zum Ändern des Wertes der Option REMOTE_OBJECT verwenden, treten Fehler auf, wenn die Struktur des neuen Objekts nicht der Struktur des Originalobjekts entspricht.</p> <p>Für Entrez: Gibt den Namen des Entrez-Objektyps an, der dem Kurznamen zugeordnet ist. Dieser Name legt das Schema und die NCBI-Datenbank für den Kurznamen sowie seine Beziehung zu anderen Kurznamen fest. Bei diesem Namen muss die Groß-/Kleinschreibung beachtet werden.</p>	
SOAPACTION	<p>Das Attribut URI SOAPACTION des WSDL-Formats (WSDL = Web Service Description Language). Diese Option wird für den Rootkurznamen benötigt. Bei anderen Kurznamen ist sie nicht zulässig.</p>	



Tabelle 152. Kurznamenoptionen und zugehörige Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
SORTED	<p>Gibt an, ob die Datenquellendatei sortiert oder unsortiert ist. Für diese Option können die Werte Y, y, n oder N angegeben werden.</p> <p>Sortierte Datenquellen müssen in aufsteigender Reihenfolge entsprechend der Sortierfolge der aktuellen Ländereinstellung sortiert werden, die in den Einstellungen in der Kategorie LC_COLLATE für die Unterstützung in der Landessprache definiert ist.</p> <p>Wenn Sie angeben, dass die Datenquelle sortiert ist, müssen Sie die Option VALIDATE_DATA_FILE auf Y setzen.</p>	N
SORTFIELD	<p>Gibt den Namen eines Feldes an, nach dem die Suchergebnisse sortiert werden sollen. Der Standardwert DOC_RANK gibt ein Feld an, das von Extended Search zur Feststellung der Relevanz eines Ergebnisdokuments verwendet wird. Wenn Sie einen anderen Feldnamen angeben, muss dieser Name in den durchsuchten Quellenobjekten vorhanden sein.</p>	DOC_RANK
SORTORDER	<p>Gibt eine Sortierreihenfolge für die Rückgabe der Suchergebnisse in aufsteigender (A) oder absteigender (D = Descending) Reihenfolge an.</p>	A
STREAMING	<p>Gibt an, ob das XML-Quelldokument zu Verarbeitungszwecken in logische Fragmente aufgeteilt werden soll. Die Fragmente entsprechen dem Knoten, der mit dem XPath-Ausdruck des Kurznamens übereinstimmt. Der Wrapper führt anschließend fragmentweise die Syntaxanalyse und die Verarbeitung der XML-Quelldaten durch. Dieser Typ der Syntaxanalyse reduziert die Speicherbelegung auf ein Minimum. Diese Option wird ausschließlich für den Rootkurznamen angegeben.</p> <p>Sie können das Streaming für jedes XML-Quelldokument (FILE, DIRECTORY, URI oder COLUMN) angeben. Diese Option ist nur für Spalten des Rootkurznamens (d. h. des Kurznamens für die Elemente der obersten Hierarchieebene des XML-Dokuments) zulässig.</p> <p>Gültige Werte sind:</p> <p>Y Die XML-Dokumente werden syntaktisch analysiert.</p> <p>N Die XML-Dokumente werden nicht syntaktisch analysiert.</p> <p>Setzen Sie den Parameter STREAMING nicht auf YES, wenn der Parameter VALIDATE auf YES gesetzt ist. Wenn Sie beide Parameter auf YES setzen, wird eine Fehlermeldung ausgegeben.</p>	N

Tabelle 152. Kurznamenoptionen und zugehörige Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
TEMPLATE	<p>Für WebSphere Business Integration: Das Kurznamenschablonenfragment, das zur Erstellung eines XML-Eingabedokuments verwendet wird. Das Fragment muss der angegebenen Schablonensyntax entsprechen.</p> <p>Für Web-Services: Das Kurznamenschablonenfragment, das zum Erstellen einer SOAP-Anforderung verwendet wird. Das Fragment muss der angegebenen Schablonensyntax entsprechen.</p>	
TOTALMAXHIT	Ein INTEGER-Wert, der die maximale Anzahl von Ergebnissen angibt, die aus allen durchsuchten Quellen zurückgegeben werden kann. Der Wrapper kombiniert diese Ergebnisse zu einer einzigen Ergebnismenge.	50
TIMEOUT	<p>Für Extended Search: Ein INTEGER-Wert, der die Anzahl der Sekunden angibt, die auf eine Antwort eines Servers gewartet wird, bevor für die Anforderung eine Zeitlimitüberschreitung ausgegeben wird.</p> <p>Für BLAST und HMMER: Die maximal zulässige Zeit (in Minuten), die der Wrapper auf die Ergebnisse des Dämons wartet.</p>	<p>Für Extended Search: 30.</p> <p>Für BLAST und HMMER: 60.</p>
URL	Die URL für den Web-Service-Endpunkt. Diese Option wird für den Rootkurznamen benötigt. Bei anderen Kurznamen ist sie nicht zulässig. Die unterstützten Protokolle sind HTTP und HTTPS.	
VALIDATE	<p>Gibt an, ob das XML-Quelldokument vor der Extraktion der XML-Daten ausgewertet werden soll. Wenn für diese Option die Einstellung YES definiert wird, prüft die Kurznamenoption, ob die Struktur des Quelldokuments mit einem XML-Schema oder einer Dokumenttypdefinition (DTD) übereinstimmt. Diese Option ist nur für Spalten des Rootkurznamens (d. h. des Kurznamens für die Elemente der obersten Hierarchieebene des XML-Dokuments) zulässig.</p> <p>Das XML-Quelldokument wird nicht ausgewertet, wenn der XML-Wrapper die XML-Schemadatei bzw. DTD-Datei (.xsd oder .dtd) nicht lokalisieren kann. DB2 UDB gibt keine Fehlermeldung aus, wenn die Auswertung nicht durchgeführt wird. Aus diesem Grund müssen Sie sicherstellen, dass die XML-Schemadatei bzw. DTD-Datei an der im XML-Quelldokument angegebenen Position vorhanden ist.</p> <p>Setzen Sie den Parameter VALIDATE nicht auf YES, wenn der Parameter STREAMING auf YES gesetzt ist. Wenn Sie beide Parameter auf YES setzen, wird eine Fehlermeldung ausgegeben.</p>	NO

Tabelle 152. Kurznamenoptionen und zugehörige Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
VALIDATE_DATA_FILE	Bei sortierten Dateien gibt diese Option an, ob der Wrapper die Sortierung der Spalten in aufsteigender Reihenfolge sowie das Vorhandensein von Nullwertschlüsseln überprüft. Als einzige Werte für diese Option sind Y oder N zulässig. Die Überprüfung wird einmalig bei der Registrierung durchgeführt. Diese Option ist nicht zulässig, wenn für den Dateipfad die Kurznamenspaltenoption DOCUMENT verwendet wird.	N
VERTICAL_TABLE	Gibt das Darstellungsformat für Suchergebnisse an. Wenn Sie hier YES angeben, gibt Extended Search zusätzlich zu den benutzerdefinierten Spalten alle Felder zurück, für die die Möglichkeit zur Rückgabe von Daten konfiguriert ist. Der Wrapper speichert die Ergebnisse in der Kurznamentabelle als vertikal angeordnete Liste mit Spaltennamen.	NO
XPATH	Gibt den XPATH-Ausdruck an, der die Elemente zur Darstellung der einzelnen Tupel identifiziert. Die Kurznamenoption XPATH für einen untergeordneten Kurznamen wird im Kontext des Pfads ausgewertet, der in der Kurznamenoption XPATH des zugehörigen übergeordneten Kurznamens angegeben wird. Dieser XPATH-Ausdruck wird als Kontext für die Auswertung von Spaltenwerten benutzt, die in den XPATH-Kurznamenspaltenoptionen angegeben sind.  Für XML: Geben Sie in einem XPATH-Ausdruck kein Namensbereichspräfix an. Der XML-Wrapper bietet keine Namensbereichsunterstützung.	
XPATH_EVAL_TIME	Gibt die Zeit (in Millisekunden) an, die zur Auswertung des XPath-Ausdrucks des Kurznamens und zur Lokalisierung des ersten Elements benötigt wird. Sie können die Optionen XPATH_EVAL_TIME, INSTANCE_PARSE_TIME und NEXT_TIME ändern, um Abfragen in umfangreichen oder komplexen XML-Quellenstrukturen zu optimieren. Diese Option ist für Rootkurznamen und Nicht-Rootkurznamen zulässig. Die angegebene Zahl kann ein Integer- oder ein Dezimalwert sein.	1

## Anhang G. Kurznamenspaltenoptionen für Systeme zusammengesessener Datenbanken

In der Anweisung CREATE NICKNAME oder ALTER NICKNAME können Sie Spalteninformationen angeben, indem Sie Parameter verwenden, die als Kurznamenspaltenoptionen bezeichnet werden.

In der folgenden Tabelle werden die Kurznamenspaltenoptionen für die verschiedenen Datenquellen aufgeführt.

Tabelle 153. Verfügbare Kurznamenspaltenoptionen

Datenquelle	ALL_VALUES	DEFAULT	DELIMITER	DOCUMENT	ESCAPE_INPUT	FOREIGN_KEY	INDEX	IS_REPEATING	NUMERIC_STRING	PRIMARY_KEY	REMOTE_NAME	SOAPACTIONCOLUMN	TEMPLATE	URLCOLUMN	VARCHAR_NO_TRAILING_BLANKS	XPATH
BLAST		X	X				X									
DB2 Universal Database für iSeries									X							
DB2 Universal Database für z/OS und OS/390									X							
DB2 Universal Database für VM und VSE									X							
DB2 Universal Database für Linux, UNIX und Windows									X							
Documentum	X		X					X			X					
Informix									X							
Microsoft SQL Server									X							
ODBC									X							
OLE DB									X							
Oracle									X						X	
Sybase									X							
Dateien mit Tabellenstruktur				X												
Teradata									X							

Tabelle 153. Verfügbare Kurznamenspaltenoptionen (Forts.)

Datenquelle	ALL_VALUES	DEFAULT	DELIMITER	DOCUMENT	ESCAPE_INPUT	FOREIGN_KEY	INDEX	IS_REPEATING	NUMERIC_STRING	PRIMARY_KEY	REMOTE_NAME	SOAPACTIONCOLUMN	TEMPLATE	URLCOLUMN	VARCHAR_NO_TRAILING_BLANKS	XPATH
WebSphere Business Integration					X	X				X			X			X
Web-Services					X	X				X		X	X	X		X
XML				X		X				X						X

Tabelle 154. Spaltenoptionen und deren Einstellungen

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
ALL_VALUES	Gibt an, dass alle Werte eines sich wiederholenden Attributs zurückgegeben werden, wobei die einzelnen Werte durch den angegebenen Begrenzer voneinander getrennt werden. Wenn diese Option fehlt oder wenn hier N angegeben wurde, wird lediglich der letzte Wert eines sich wiederholenden Attributs zurückgegeben. Die Option ALL_VALUES kann nur für Spalten vom Typ VARCHAR angegeben werden, für die die Option IS_REPEATING auf 'Y' eingestellt ist. (Die Option ist hingegen ungültig, wenn für IS_REG_TABLE = 'Y' definiert wurde.)	
DEFAULT	Gibt einen neuen Standardwert für die folgenden Eingabespalten mit festgelegtem Format an: <ul style="list-style-type: none"> <li>• E_value</li> <li>• QueryStrands</li> <li>• GapAlign</li> <li>• NMisMatchPenalty</li> <li>• NMatchReward</li> <li>• Matrix</li> <li>• FilterSequence</li> <li>• NumberOfAlignments</li> <li>• GapCost</li> <li>• ExtendedGapCost</li> <li>• WordSize</li> <li>• ThresholdEx</li> </ul> <p>Dieser neue Wert überschreibt die vordefinierten Standardwerte. Der neue Standardwert muss denselben Typ wie der für eine bestimmte Spalte angegebene Wert aufweisen.</p>	

Tabelle 154. Spaltenoptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
DELIMITER	<p>Für Documentum: Gibt die Begrenzerzeichenfolge an, die bei der Verknüpfung mehrerer Werte eines sich wiederholenden Attributs benutzt werden soll. Der Begrenzer kann aus einem oder mehreren Zeichen bestehen. Diese Option ist nur für Attribute von Objekten zulässig, die den Datentyp VARCHAR aufweisen und für die die Option IS_REPEATING auf Y eingestellt ist.</p> <p>Für BLAST: Die Begrenzungszeichen zur Festlegung des Endpunktes der Definitionszeileninformationen für die Spalte, in der diese Option aufgeführt ist. Wenn im Wert dieser Option mehr als ein Zeichen definiert ist, dann wird das erste Vorkommen eines dieser Zeichen als das Ende der in diesem Feld enthaltenen Daten interpretiert. Standardmäßig ist dies am Zeilenende der Fall. Diese Option ist erforderlich, wenn nicht in der letzten angegebenen Spalte die restlichen Informationen der Definitionszeile enthalten sein sollen.</p>	<p>Für Documentum: Als Begrenzer wird standardmäßig ein Komma verwendet.</p> <p>Für BLAST: Als Begrenzer wird standardmäßig das Zeilenende verwendet.</p>

Tabelle 154. Spaltenoptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
DOCUMENT	<p>Für Dateien mit Tabellenstruktur: Gibt den Typ der Datei mit Tabellenstruktur an. Dieser Wrapper unterstützt bei dieser Option nur den Wert FILE. Pro Kurzname kann bei der Option DOCUMENT nur eine Spalte angegeben werden. Die Spalte, die der Option DOCUMENT zugeordnet ist, muss den Datentyp VARCHAR oder CHAR aufweisen. Wenn Sie an Stelle der Kurznamenoption FILE_PATH die Kurznamenoption DOCUMENT verwenden, wird davon ausgegangen, dass die diesem Kurznamen zugeordnete Datei bei der Ausführung der Abfrage bereitgestellt wird. Wenn für die Option DOCUMENT der Wert FILE angegeben ist, wird bei der Ausführung der Abfrage der vollständige Pfad der Datei bereitgestellt, deren Schema mit der Kurznamendefinition dieses Kurznamens übereinstimmt.</p> <p>Für XML: Gibt an, dass es sich bei der aktuellen Spalte um eine DOCUMENT-Spalte handelt. Der Wert der DOCUMENT-Spalte gibt den Typ der XML-Quellendaten an, die bei der Ausführung der Abfrage für den Kurznamen bereitgestellt werden. Diese Option ist nur für Spalten des Rootkurznamens (d. h. des Kurznamens für die Elemente der obersten Hierarchieebene des XML-Dokuments) zulässig. Pro Kurzname kann bei der Option DOCUMENT nur eine Spalte angegeben werden. Die Spalte, die der Option DOCUMENT zugeordnet ist, muss den Datentyp VARCHAR aufweisen. Wenn Sie an Stelle der Kurznamenoption FILE_PATH oder DIRECTORY_PATH die Spaltenoption DOCUMENT verwenden, wird bei der Ausführung der Abfrage das diesem Kurznamen zugeordnete Dokument bereitgestellt.</p> <p>Die zulässigen Werte für die Option DOCUMENT lauten wie folgt:</p> <p><b>FILE</b> Gibt an, dass der Wert der Kurznamenspalte mit dem Pfadnamen einer Datei verknüpft ist. Die Daten aus dieser Datei werden bei der Ausführung der Abfrage bereitgestellt.</p> <p><b>DIRECTORY</b> Gibt an, dass der Wert der Kurznamenspalte mit dem Pfadnamen eines Verzeichnisses verknüpft ist, das mehrere XML-Datendateien enthält. Die XML-Daten aus den verschiedenen Dateien werden bei der Ausführung der Abfrage bereitgestellt. Die Daten sind in XML-Dateien gespeichert, die im angegebenen Verzeichnispfad abgelegt sind. Der XML-Wrapper verwendet nur Dateien mit der Erweiterung .xml, die im angegebenen Verzeichnis gespeichert sind. Alle anderen Dateien in diesem Verzeichnis werden vom XML-Wrapper hingegen ignoriert.</p> <p><b>URI</b> Gibt an, dass der Wert der Kurznamenspalte mit dem Pfadnamen einer fernen XML-Datei verknüpft ist, auf die mit einer URI verwiesen wird. Die URI-Adresse gibt die ferne Position dieser XML-Datei im Web an.</p> <p><b>COLUMN</b> Gibt an, dass das XML-Dokument in der Spalte einer relationalen Datenbank gespeichert ist.</p>	

Tabelle 154. Spaltenoptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
ELEMENT_NAME	Gibt den BioRS-Elementnamen an. Die Regelungen für die Groß-/Kleinschreibung dieses Namens hängen von den entsprechenden Regelungen für den BioRS-Server und vom Wert für die Serveroption CASE_SENSITIVE ab. Sie müssen den BioRS-Elementnamen nur angeben, wenn dieser vom Spaltennamen abweicht.	
ESCAPE_INPUT	Gibt an, ob in den XML-Eingabewerten XML-Sonderzeichen ersetzt werden sollen. Verwenden Sie diese Option, wenn XML-Fragmente (z. B. XML-Fragmente mit sich wiederholenden Elementen) als Eingabe verwendet werden sollen. Für Spalten, die die Spaltenoption ESCAPE_INPUT verwenden, muss die Spaltenoption TEMPLATE definiert werden. Als Spaltendatentyp muss VARCHAR oder CHAR benutzt werden.  Gültige Werte sind:  Y Wenn die XML-Eingabe Sonderzeichen enthält, werden diese durch die Zeichen ersetzt, die XML zur Darstellung der Eingabezeichen verwendet.  N Eingabezeichen werden genauso beibehalten, wie diese dargestellt sind.	Y
FOREIGN_KEY	Gibt an, dass der aktuelle Kurzname ein untergeordneter Kurzname ist. Außerdem wird der Name des zugehörigen übergeordneten Kurznamens angegeben. Für einen Kurznamen darf die Spaltenoption FOREIGN_KEY maximal einmal definiert sein. Beim Wert für diese Option muss die Groß-/Kleinschreibung beachtet werden. Die Tabelle, die dieser Option zugeordnet ist, enthält einen vom Wrapper generierten Schlüssel. Die Option XPATH darf für diese Spalte nicht angegeben werden. Die Spalte darf nur zur Verknüpfung von über- und untergeordneten Kurznamen eingesetzt werden.  Die Ausführung einer Anweisung CREATE NICKNAME mit der Option FOREIGN_KEY schlägt fehl, wenn der übergeordnete Kurzname einen anderen Schemanamen aufweist.  Wenn der Kurzname, auf den in der Klausel FOREIGN_KEY verwiesen wird, nicht ausdrücklich in Kleinschreibung oder gemischter Groß-/Kleinschreibung definiert wurde, indem dieser in der entsprechenden Anweisung CREATE NICKNAME in Anführungszeichen eingeschlossen wurde, müssen Sie diesen Kurznamen bei Verweisen in der Klausel FOREIGN_KEY in Großbuchstaben angeben.  Wenn diese Option für eine Spalte definiert wird, darf für die Spalte keine andere Option angegeben werden.	
INDEX	Die Ordinalzahl der Spalte, in der diese Option innerhalb der Gruppe von Definitionen aufgeführt wird. Diese Option ist erforderlich.	



Tabelle 154. Spaltenoptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
IS_INDEXED	Gibt an, ob die zugehörige Spalte indiziert ist, d. h. ob auf die Spalte in einem Vergleichselement verwiesen werden kann. Die gültigen Werte sind Y und N. Der Wert Y kann nur für Spalten angegeben werden, deren zugehöriges Element vom BioRS-Server indiziert wurde.	Bei der Erstellung eines Kurznamens wird diese Option automatisch mit dem Wert Y bei allen Spalten hinzugefügt, die einem indizierten BioRS-Element entsprechen.
IS_REPEATING	Gibt an, ob die Spalte mehrere Werte enthalten kann. Gültige Werte sind Y und N.  Für die folgenden Elemente wird nur der jeweils letzte Wert zurückgegeben: <ul style="list-style-type: none"> <li>• Sich wiederholende Attribute, deren Typ nicht VARCHAR lautet.</li> <li>• VARCHAR-Spalten, wenn für ALL_VALUES der Wert 'N' angegeben ist.</li> </ul> Um diese Einschränkung zu umgehen, können Sie eine Doppeldefinition für die Spalte mit dem sich wiederholenden Attribut erstellen.	N
NUMERIC_STRING	Gibt an, ob eine Spalte numerische Zeichenfolgen enthält.  Y Diese Spalte enthält Zeichenfolgen mit den numerischen Zeichen '0', '1', '2', .... '9'. Sie enthält keine Leerzeichen. Wenn diese Spalte nur numerische Zeichenfolgen mit abschließenden Leerzeichen enthält, geben Sie nicht Y an.  Wenn Sie für eine Spalte NUMERIC_STRING auf Y setzen, informieren Sie das Optimierungsprogramm darüber, dass diese Spalte keine Leerzeichen enthält, die sich auf die Sortierung der Spaltendaten auswirken könnten. Verwenden Sie diese Option, wenn sich die Sortierfolge einer Datenquelle von der Sortierfolge unterscheidet, die auf dem Server mit zusammengeschlossenen Datenbanken benutzt wird. Spalten, die diese Option verwenden, werden nicht auf Grund einer anderen Sortierfolge von der fernen Auswertung ausgeschlossen.  N Diese Spalte enthält entweder keine numerischen Zeichenfolgen oder es handelt sich um eine Spalte mit numerischen Zeichenfolgen, die Leerzeichen enthält.	N
PRIMARY_KEY	Gibt an, dass es sich beim aktuellen Kurznamen um einen übergeordneten Kurznamen handelt. Als Spaltentyp muss VARCHAR(16) verwendet werden. Für einen Kurznamen darf die Spaltenoption PRIMARY_KEY maximal einmal definiert sein. Der einzige zulässige Wert lautet YES. Die Spalte, die dieser Option zugeordnet ist, enthält einen Schlüssel, der vom Wrapper generiert wird. Die Option XPATH darf für diese Spalte nicht angegeben werden. Die Spalte darf nur zur Verknüpfung von über- und untergeordneten Kurznamen eingesetzt werden.  Wenn diese Option für eine Spalte definiert wird, darf für die Spalte keine andere Option angegeben werden.	

Tabelle 154. Spaltenoptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
REFERENCED_OBJECT	Diese Option ist nur für Spalten zulässig, deren BioRS-Datentyp "Reference" lautet. Sie gibt den Namen der BioRS-Datenbank an, auf die über die aktuelle Spalte verwiesen wird. Die Regelungen für die Groß-/Kleinschreibung dieses Namens hängen von den entsprechenden Regelungen für den BioRS-Server und vom Wert für die Serveroption CASE_SENSITIVE ab.	
REMOTE_NAME	Gibt den Namen des zugehörigen Documentum-Attributs bzw. der zugehörigen Documentum-Spalte an. Diese Option ordnet ferne Attribut- oder Spaltennamen bestimmten lokalen DB2 UDB-Spaltennamen zu.	Der DB2 UDB-Spaltenname.
SOAPACTIONCOLUMN	Eine Spalte zur dynamischen Angabe des Attributs URI SOAPACTION mit Hilfe des WSDL-Formats (WSDL = Web Service Description Language). Diese Option wird ausschließlich für den Rootkurznamen angegeben.  Wenn diese Option für eine Spalte definiert wird, darf für die Spalte keine andere Option angegeben werden.	
TEMPLATE	Das Spaltenschablonenfragment, das zur Erstellung des XML-Eingabedokuments verwendet wird. Das Fragment muss der angegebenen Schablonensyntax entsprechen.	
URLCOLUMN	Eine Spalte zur dynamischen Angabe der URL-Adresse für den Web-Service-Endpunkt bei der Ausführung einer Abfrage. Diese Option wird ausschließlich für den Rootkurznamen angegeben.  Wenn diese Option für eine Spalte definiert wird, darf für die Spalte keine andere Option angegeben werden.	

Tabelle 154. Spaltenoptionen und deren Einstellungen (Forts.)

Option	Beschreibung und gültige Einstellungen	Standardeinstellung
VARCHAR_NO_TRAILING_BLANKS	<p>Diese Option gilt für Datenquellen mit variablen Zeichentypen, bei denen fehlende Zeichen bis zur maximal zulässigen Länge bei Vergleichsoperationen nicht mit folgenden Leerzeichen aufgefüllt werden.</p> <p>Einige Datenquellen wie z. B. Oracle verfügen nicht über eine Vergleichssemantik, bei der zum Auffüllen Leerzeichen verwendet werden und die dieselben Ergebnisse zurückgibt wie die Vergleichssemantik von DB2 UDB für Linux, UNIX und Windows. Aktivieren Sie diese Option, wenn sie nur auf eine bestimmte VARCHAR- oder VARCHAR2-Spalte in einem Datenquellenobjekt angewendet werden soll.</p> <p>Y Diese VARCHAR-Spalten enthalten keine abschließenden Leerzeichen, oder diese Datenquelle verfügt über eine Vergleichssemantik mit Leerzeichen als Füllzeichen, die mit der Vergleichssemantik des Servers mit zusammengeschlossenen Datenbanken übereinstimmt.</p> <p>Der Server mit zusammengeschlossenen Datenbanken sendet Zeichenvergleichsoperationen zur Verarbeitung an die Datenquelle.</p> <p>N Diese VARCHAR-Spalten enthalten abschließende Leerzeichen, und die Datenquelle verfügt über eine Vergleichssemantik mit Leerzeichen als Füllzeichen, die von der Vergleichssemantik des Servers mit zusammengeschlossenen Datenbanken abweicht.</p> <p>Der Server mit zusammengeschlossenen Datenbanken verarbeitet Zeichenvergleichsoperationen, wenn ein Ausgleich für äquivalente Semantiktypen nicht möglich ist. Dies gilt z. B. beim erneuten Schreiben des Vergleichselements.</p>	N für betroffene Datenquellen
XPATH	Gibt den XPath-Ausdruck im XML-Dokument an, der die Daten enthält, die dieser Spalte entsprechen. Der Wrapper wertet den XPath-Ausdruck aus, nachdem dieser von der Anweisung CREATE NICKNAME über die Kurznamenoption XPATH angewendet wurde.	

**Zugehörige Konzepte:**

- „Pushdown-Analyse“ in *Systeme zusammengeschlossener Datenbanken*

**Zugehörige Tasks:**

- „Globale Optimierung“ in *Systeme zusammengeschlossener Datenbanken*

## Anhang H. Standardmäßige progressive Datentypzuordnungen

Zwischen den Datentypen von Datenquellen und zusammengeschlossenen Datenbanken gibt es zwei Typen von Zuordnungen. Dies ist zum Einen die progressive und zum Anderen die regressive Typzuordnung. In einer *progressiven Typzuordnung* erfolgt die Zuordnung von einem fernen Typ zu einem vergleichbaren lokalen Typ.

Sie können eine Standardtypzuordnung überschreiben oder unter Verwendung der Anweisung CREATE TYPE MAPPING eine neue Typzuordnung erstellen.

Diese Zuordnungen sind für alle unterstützten Versionen gültig, falls nichts anderes angemerkt ist.

Für alle standardmäßigen progressiven Datentypzuordnungen zwischen einer Datenquelle und DB2 für Linux, UNIX und Windows lautet das DB2-Schema des Systems zusammengeschlossener Datenbanken SYSIBM.

In den folgenden Tabellen sind die standardmäßigen progressiven Typzuordnungen zwischen Datentypen von DB2 für Linux, UNIX und Windows und Datentypen von Datenquellen aufgeführt.

### Datenquellen von DB2 für z/OS und OS/390

Tabelle 155. Standardmäßige progressive Typzuordnungen von DB2 für z/OS und OS/390 (keine vollständige Anzeige der Spalten)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	1	254	-	-	-	-	CHAR	-	0	N
CHAR	255	32672	-	-	-	-	VARCHAR	-	0	N
CHAR	1	254	-	-	Y	-	CHAR	-	0	Y
CHAR	255	32672	-	-	Y	-	VARCHAR	-	0	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
FLOAT	4	-	-	-	-	-	REAL	-	-	-
FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-

Tabelle 155. Standardmäßige progressive Typzuordnungen von DB2 für z/OS und OS/390 (keine vollständige Anzeige der Spalten) (Forts.)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
ROWID	-	-	-	-	Y	-	VARCHAR	40	-	Y
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	Y	-	VARCHAR	-	0	Y
VARG	1	16336	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	0	N

## Datenquellen von DB2 für iSeries

Tabelle 156. Standardmäßige progressive Typzuordnungen von DB2 für iSeries (keine vollständige Anzeige der Spalten)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	1	254	-	-	-	-	CHAR	-	0	N
CHAR	255	32672	-	-	-	-	VARCHAR	-	0	N
CHAR	1	254	-	-	Y	-	CHAR	-	0	Y
CHAR	255	32672	-	-	Y	-	VARCHAR	-	0	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-

Tabelle 156. Standardmäßige progressive Typzuordnungen von DB2 für iSeries (keine vollständige Anzeige der Spalten) (Forts.)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
I FLOAT	4	-	-	-	-	-	REAL	-	-	-
I FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
GRAPHIC	128	16336	-	-	-	-	VARGRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
NUMERIC	-	-	-	-	-	-	DECIMAL	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	Y	-	VARCHAR	-	0	Y
VARG	1	16336	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	0	N

## Datenquellen von DB2 Server für VM und VSE

Tabelle 157. Standardmäßige progressive Typzuordnungen von DB2 Server für VM und VSE (keine vollständige Anzeige der Spalten)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	1	254	-	-	-	-	CHAR	-	0	N
CHAR	1	254	-	-	Y	-	CHAR	-	0	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBAHW	-	-	-	-	-	-	SMALLINT	-	0	-
DBAINT	-	-	-	-	-	-	INTEGER	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
FLOAT	4	-	-	-	-	-	REAL	-	-	-
FLOAT	8	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	0	N
VARCHAR	1	32672	-	-	Y	-	VARCHAR	-	0	Y
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPH	1	16336	-	-	-	-	VARGRAPHIC	-	0	N

## Datenquellen von DB2 für Linux, UNIX und Windows

Tabelle 158. Standardmäßige progressive Typzuordnungen von DB2 für Linux, UNIX und Windows (keine vollständige Anzeige der Spalten)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BIGINT	-	-	-	-	-	-	BIGINT	-	0	-
I BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	-	-	-	-	-	-	CHAR	-	0	N
CHAR	-	-	-	-	Y	-	CHAR	-	0	Y
I CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
I DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	-	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
I LONGVAR	-	-	-	-	N	-	CLOB	-	-	-
I LONGVAR	-	-	-	-	Y	-	BLOB	-	-	-
I LONGVARG	-	-	-	-	-	-	DBCLOB	-	-	-
REAL	-	-	-	-	-	-	REAL	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	0	-
TIME	-	-	-	-	-	-	TIME	-	0	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
TIMESTMP	-	-	-	-	-	-	TIMESTAMP	-	0	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	0	N
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	0	Y
VARGRAPH	-	-	-	-	-	-	VARGRAPHIC	-	0	N
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	0	N



## Informix-Datenquellen

Tabelle 159. Standardmäßige progressive Typzuordnungen von Informix (keine vollständige Anzeige der Spalten)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	2147483647	-	-
BOOLEAN	-	-	-	-	-	-	CHARACTER	1	-	-
BYTE	-	-	-	-	-	-	BLOB	2147483647	-	-
CHAR	1	254	-	-	-	-	CHARACTER	-	-	-
CHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
CLOB	-	-	-	-	-	-	CLOB	2147483647	-	-
DATE	-	-	-	-	-	-	DATE	4	-	-
DATETIME	0	4	0	4	-	-	DATE	4	-	-
DATETIME	6	10	6	10	-	-	TIME	3	-	-
DATETIME	0	4	6	15	-	-	TIMESTAMP	10	-	-
DATETIME	6	10	11	15	-	-	TIMESTAMP	10	-	-
DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
DECIMAL	32	130	-	-	-	-	DOUBLE	8	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	8	-	-
INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
INTERVAL	-	-	-	-	-	-	VARCHAR	25	-	-
INT8	-	-	-	-	-	-	BIGINT	19	0	-
LVARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
MONEY	1	31	0	31	-	-	DECIMAL	-	-	-
MONEY	32	32	-	-	-	-	DOUBLE	8	-	-
NCHAR	1	254	-	-	-	-	CHARACTER	-	-	-
NCHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
NVARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
REAL	-	-	-	-	-	-	REAL	4	-	-
SERIAL	-	-	-	-	-	-	INTEGER	4	-	-
SERIAL8	-	-	-	-	-	-	BIGINT	-	-	-
SMALLFLOAT	-	-	-	-	-	-	REAL	4	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
TEXT	-	-	-	-	-	-	CLOB	2147483647	-	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-

Tabelle 159. Standardmäßige progressive Typzuordnungen von Informix (keine vollständige Anzeige der Spalten) (Forts.)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
-----------------	------------------	------------------	--------------------	--------------------	-----------------	-----------------------	--------------------	------------------	-----------------	--------------------

**Anmerkungen:**

- Für den Informix-Datentyp DATETIME verwendet der DB2-Server mit zusammengeschlossenen Datenbanken unter UNIX und Windows das Informix-Qualifikationsmerkmal der oberen Ebene als REMOTE\_LENGTH und das Informix-Qualifikationsmerkmal der unteren Ebene als REMOTE\_SCALE.

Die Informix-Qualifikationsmerkmale sind die Konstanten "TU\_", die in der Informix Client SDK-Datei `datatime.h` definiert sind. Es gibt folgende Konstanten:

0 = YEAR	8 = MINUTE	13 = FRACTION(3)
2 = MONTH	10 = SECOND	14 = FRACTION(4)
4 = DAY	11 = FRACTION(1)	15 = FRACTION(5)
6 = HOUR	12 = FRACTION(2)	

## Microsoft SQL Server-Datenquellen

Tabelle 160. Standardmäßige progressive Datentypzuordnungen für Microsoft SQL Server

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
bigint <sup>4</sup>	-	-	-	-	-	-	BIGINT	-	-	-
binary	1	254	-	-	-	-	CHARACTER	-	-	Y
binary	255	8000	-	-	-	-	VARCHAR	-	-	Y
bit	-	-	-	-	-	-	SMALLINT	2	-	-
char	1	254	-	-	-	-	CHAR	-	-	N
char	255	8000	-	-	-	-	VARCHAR	-	-	N
datetime	-	-	-	-	-	-	TIMESTAMP	10	-	-
datetimen	-	-	-	-	-	-	TIMESTAMP	10	-	-
decimal	1	31	0	31	-	-	DECIMAL	-	-	-
decimal	32	38	0	38	-	-	DOUBLE	-	-	-

Tabelle 160. Standardmäßige progressive Datentypzuordnungen für Microsoft SQL Server (Forts.)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
decimaln	1	31	0	31	-	-	DECIMAL	-	-	-
decimaln	32	38	0	38	-	-	DOUBLE	-	-	-
DUMMY65 <sup>1</sup>	1	38	-84	127	-	-	DOUBLE	-	-	-
DUMMY2000 <sup>3</sup>	1	38	-84	127	-	-	DOUBLE	-	-	-
float	-	8	-	-	-	-	DOUBLE	8	-	-
floatn	-	8	-	-	-	-	DOUBLE	8	-	-
float	-	4	-	-	-	-	REAL	4	-	-
floatn	-	4	-	-	-	-	REAL	4	-	-
image	-	-	-	-	-	-	BLOB	2147483647	-	Y
int	-	-	-	-	-	-	INTEGER	4	-	-
intn	-	-	-	-	-	-	INTEGER	4	-	-
money	-	-	-	-	-	-	DECIMAL	19	4	-
moneyn	-	-	-	-	-	-	DECIMAL	19	4	-
nchar	1	127	-	-	-	-	CHAR	-	-	N
nchar	128	4000	-	-	-	-	VARCHAR	-	-	N
numeric	1	31	0	31	-	-	DECIMAL	-	-	-
numeric	32	38	0	38	-	-	DOUBLE	8	-	-
numericn	32	38	0	38	-	-	DOUBLE	-	-	-
numericn	1	31	0	31	-	-	DECIMAL	-	-	-
ntext <sup>2</sup>	-	-	-	-	-	-	CLOB	2147483647	-	Y
nvarchar	1	4000	-	-	-	-	VARCHAR	-	-	N
real	-	-	-	-	-	-	REAL	4	-	-
smallint	-	-	-	-	-	-	SMALLINT	2	-	-
smalldatetime	-	-	-	-	-	-	TIMESTAMP	10	-	-
smallmoney	-	-	-	-	-	-	DECIMAL	10	4	-
smallmoneyn	-	-	-	-	-	-	DECIMAL	10	4	-
SQL_BIGINT	-	-	-	-	-	-	DECIMAL	-	-	-
SQL_BIGINT <sup>4</sup>	-	-	-	-	-	-	BIGINT	-	-	-
SQL_BINARY	1	254	-	-	-	-	CHARACTER	-	-	Y
SQL_BINARY	255	8000	-	-	-	-	VARCHAR	-	-	Y
SQL_BIT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_CHAR	1	254	-	-	-	-	CHAR	-	-	N

Tabelle 160. Standardmäßige progressive Datentypzuordnungen für Microsoft SQL Server (Forts.)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
SQL_CHAR	255	8000	-	-	-	-	VARCHAR	-	-	N
SQL_DATE	-	-	-	-	-	-	DATE	4	-	-
SQL_DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_DECIMAL	32	38	0	38	-	-	DOUBLE	8	-	-
SQL_DECIMAL	32	32	0	31	-	-	DOUBLE	8	-	-
SQL_DOUBLE	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_FLOAT	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_GUID <sup>2</sup>	1	4000	-	-	Y	-	VARCHAR	16	-	Y
SQL_INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
SQL_LONGVARCHAR	-	-	-	-	-	-	CLOB	2147483647	-	N
SQL_LONGVARBINARY	-	-	-	-	-	-	BLOB	-	-	Y
SQL_NUMERIC	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_REAL	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_TIME	-	-	-	-	-	-	TIME	3	-	-
SQL_TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	10	-	-
SQL_TINYINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_VARBINARY	1	8000	-	-	-	-	VARCHAR	-	-	Y
SQL_VARCHAR	1	8000	-	-	-	-	VARCHAR	-	-	N
text	-	-	-	-	-	-	CLOB	-	-	N
timestamp	-	-	-	-	-	-	VARCHAR	8	-	Y
tinyint	-	-	-	-	-	-	SMALLINT	2	-	-
uniqueidentifier <sup>2</sup>	1	4000	-	-	Y	-	VARCHAR	16	-	Y
varbinary	1	8000	-	-	-	-	VARCHAR	-	-	Y
varchar	1	8000	-	-	-	-	VARCHAR	-	-	N

**Anmerkungen:**

- | 1. Diese Typzuordnung ist nur für Microsoft SQL Server Version 6.5 gültig.
- | 2. Diese Typzuordnung ist nur für Microsoft SQL Server Version 7 und Version 2000 gültig.
- | 3. Diese Typzuordnung ist nur für Windows 2000 gültig.
- | 4. Diese Typzuordnung ist nur für Microsoft SQL Server Version 2000 gültig.

## ODBC-Datenquellen

Tabelle 161. Standardmäßige progressive Typzuordnungen von ODBC (keine vollständige Anzeige der Spalten)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
SQL_BIGINT	-	-	-	-	-	-	BIGINT	8	-	-
SQL_BINARY	1	254	-	-	-	-	CHARACTER	-	-	Y
SQL_BINARY	255	32672	-	-	-	-	VARCHAR	-	-	Y
SQL_BIT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_CHAR	1	254	-	-	-	-	CHAR	-	-	N
SQL_CHAR	255	32672	-	-	-	-	VARCHAR	-	-	N
SQL_DECIMAL	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_DECIMAL	32	38	0	38	-	-	DOUBLE	8	-	-
SQL_DOUBLE	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_FLOAT	-	-	-	-	-	-	DOUBLE	8	-	-
SQL_INTEGER	-	-	-	-	-	-	INTEGER	4	-	-
SQL_LONGVARCHAR	-	-	-	-	-	-	CLOB	2147483647	-	N
SQL_LONGVARBINARY	-	-	-	-	-	-	BLOB	-	-	Y
SQL_NUMERIC	1	31	0	31	-	-	DECIMAL	-	-	-
SQL_NUMERIC	32	32	0	31	-	-	DOUBLE	8	-	-
SQL_REAL	-	-	-	-	-	-	REAL	4	-	-
SQL_SMALLINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_TYPE_DATE	-	-	-	-	-	-	DATE	4	-	-
SQL_TYPE_TIME	-	-	-	-	-	-	TIME	3	-	-
SQL_TYPE_TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	10	-	-
SQL_TINYINT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_VARBINARY	1	32672	-	-	-	-	VARCHAR	-	-	Y
SQL_VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	N
SQL_WCHAR	1	127	-	-	-	-	CHAR	-	-	N
SQL_WCHAR	128	16336	-	-	-	-	VARCHAR	-	-	N
SQL_WVARCHAR	1	16336	-	-	-	-	VARCHAR	-	-	N
SQL_WLONGVARCHAR	-	1073741823	-	-	-	-	CLOB	2147483647	-	N

## Oracle NET8-Datenquellen

Tabelle 162. Standardmäßige progressive Datentypzuordnungen für Oracle NET8

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BLOB	0	0	0	0	-	\0	BLOB	2147483647	0	Y
CHAR	1	254	0	0	-	\0	CHAR	0	0	N
CHAR	255	2000	0	0	-	\0	VARCHAR	0	0	N
CLOB	0	0	0	0	-	\0	CLOB	2147483647	0	N
DATE	0	0	0	0	-	\0	TIMESTAMP	0	0	N
FLOAT	1	126	0	0	-	\0	DOUBLE	0	0	N
LONG	0	0	0	0	-	\0	CLOB	2147483647	0	N
LONG RAW	0	0	0	0	-	\0	BLOB	2147483647	0	Y
MLSLABEL	0	0	0	0	-	\0	VARCHAR	255	0	N
NUMBER	1	38	-84	127	-	\0	DOUBLE	0	0	N
NUMBER	1	31	0	31	-	>=	DECIMAL	0	0	N
NUMBER	1	4	0	0	-	\0	SMALLINT	0	0	N
NUMBER	5	9	0	0	-	\0	INTEGER	0	0	N
NUMBER	-	10	0	0	-	\0	DECIMAL	0	0	N
RAW	1	2000	0	0	-	\0	VARCHAR	0	0	Y
ROWID	0	0	0	NULL	-	\0	CHAR	18	0	N
TIMESTAMP <sup>1</sup>	-	-	-	-	-	-	TIMESTAMP	10	-	-
VARCHAR2	1	4000	0	0	-	\0	VARCHAR	0	0	N

### Anmerkungen:

1. Diese Typzuordnung ist nur für Client- und Serverkonfigurationen unter Oracle 9i (oder einer späteren Version dieses Produkts) zulässig.

## Sybase-Datenquellen

Tabelle 163. Standardmäßige progressive Datentypzuordnungen für Sybase CTLIB

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
binary	1	254	-	-	-	-	CHAR	-	-	Y
binary	255	16384	-	-	-	-	VARCHAR	-	-	Y
bit	-	-	-	-	-	-	SMALLINT	-	-	-
char	1	254	-	-	-	-	CHAR	-	-	N
char	255	16384	-	-	-	-	VARCHAR	-	-	N
char null (siehe varchar)										
datetime	-	-	-	-	-	-	TIMESTAMP	-	-	-
datetimn	-	-	-	-	-	-	TIMESTAMP	-	-	-
decimal	1	31	0	31	-	-	DECIMAL	-	-	-
decimal	32	38	0	38	-	-	DOUBLE	-	-	-
decimaln	1	31	0	31	-	-	DECIMAL	-	-	-
decimaln	32	38	0	38	-	-	DOUBLE	-	-	-
float	-	4	-	-	-	-	REAL	-	-	-
float	-	8	-	-	-	-	DOUBLE	-	-	-
floatn	-	4	-	-	-	-	REAL	-	-	-
floatn	-	8	-	-	-	-	DOUBLE	-	-	-
image	-	-	-	-	-	-	BLOB	-	-	-
int	-	-	-	-	-	-	INTEGER	-	-	-
intn	-	-	-	-	-	-	INTEGER	-	-	-
money	-	-	-	-	-	-	DECIMAL	19	4	-
moneyn	-	-	-	-	-	-	DECIMAL	19	4	-
nchar	1	254	-	-	-	-	CHAR	-	-	N
nchar	255	16384	-	-	-	-	VARCHAR	-	-	N
nchar null (siehe nvarchar)										
numeric	1	31	0	31	-	-	DECIMAL	-	-	-
numeric	32	38	0	38	-	-	DOUBLE	-	-	-
numericn	1	31	0	31	-	-	DECIMAL	-	-	-
numericn	32	38	0	38	-	-	DOUBLE	-	-	-
nvarchar	1	16384	-	-	-	-	VARCHAR	-	-	N
real	-	-	-	-	-	-	REAL	-	-	-

Tabelle 163. Standardmäßige progressive Datentypzuordnungen für Sybase CTLIB (Forts.)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
smalldatetime	-	-	-	-	-	-	TIMESTAMP	-	-	-
smallint	-	-	-	-	-	-	SMALLINT	-	-	-
smallmoney	-	-	-	-	-	-	DECIMAL	10	4	-
sysname	1	254	-	-	-	-	CHAR	-	-	N
text	-	-	-	-	-	-	CLOB	-	-	-
timestamp	-	-	-	-	-	-	VARCHAR	8	-	Y
tinyint	-	-	-	-	-	-	SMALLINT	-	-	-
unichar <sup>1</sup>	1	254	-	-	-	-	CHAR	-	-	N
unichar <sup>1</sup>	255	16384	-	-	-	-	VARCHAR	-	-	N
unichar null (siehe univarchar)										
univarchar <sup>1</sup>	1	16384	-	-	-	-	VARCHAR	-	-	N
varbinary	1	16384	-	-	-	-	VARCHAR	-	-	Y
varchar	1	16384	-	-	-	-	VARCHAR	-	-	N

**Anmerkungen:**

1. Zulässig für zusammengeschlossene Datenbanken ohne Unicode-Unterstützung.

## Teradata-Datenquellen

Tabelle 164. Standardmäßige progressive Typzuordnungen von Teradata (keine vollständige Anzeige der Spalten)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BYTE	1	254	-	-	-	-	CHAR	-	-	Y
BYTE	255	32672	-	-	-	-	VARCHAR	-	-	Y
BYTE	32673	64000	-	-	-	-	BLOB	-	-	-



| Tabelle 164. Standardmäßige progressive Typzuordnungen von Teradata (keine vollständige Anzeige der  
| Spalten) (Forts.)

REMOTE_TYPENAME	REMOTE_LOWER_LEN	REMOTE_UPPER_LEN	REMOTE_LOWER_SCALE	REMOTE_UPPER_SCALE	REMOTE_BIT_DATA	REMOTE_DATA_OPERATORS	FEDERATED_TYPENAME	FEDERATED_LENGTH	FEDERATED_SCALE	FEDERATED_BIT_DATA
BYTEINT	-	-	-	-	-	-	SMALLINT	-	-	-
CHAR	1	254	-	-	-	-	CHARACTER	-	-	-
CHAR	255	32672	-	-	-	-	VARCHAR	-	-	-
CHAR	32673	64000	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	-	-
DECIMAL	1	18	0	18	-	-	DECIMAL	-	-	-
DOUBLE PRECISION	-	-	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	1	127	-	-	-	-	GRAPHIC	-	-	-
GRAPHIC	128	16336	-	-	-	-	VARGRAPHIC	-	-	-
GRAPHIC	16337	32000	-	-	-	-	DBCLOB	-	-	-
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
INTERVAL	-	-	-	-	-	-	CHAR	-	-	-
NUMERIC	1	18	0	18	-	-	DECIMAL	-	-	-
REAL	-	-	-	-	-	-	DOUBLE	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	-	-
VARBYTE	1	32762	-	-	-	-	VARCHAR	-	-	Y
VARBYTE	32763	64000	-	-	-	-	BLOB	-	-	-
VARCHAR	1	32672	-	-	-	-	VARCHAR	-	-	-
VARCHAR	32673	64000	-	-	-	-	CLOB	-	-	-
VARGRAPHIC	1	16336	-	-	-	-	VARGRAPHIC	-	-	-
VARGRAPHIC	16337	32000	-	-	-	-	DBCLOB	-	-	-

**Zugehörige Konzepte:**

- „Progressive und regressive Datentypzuordnungen“ in *Systeme zusammenschlossener Datenbanken*

**Zugehörige Referenzen:**

- „Ändern langer Datentypen in VARCHAR-Datentypen“ in *Systeme zusammenschlossener Datenbanken*
- „Standardmäßige progressive Typzuordnungen in Unicode - NET8-Wrapper“ in *Systeme zusammenschlossener Datenbanken*
- „Standardmäßige progressive Typzuordnungen in Unicode - Sybase-Wrapper“ in *Systeme zusammenschlossener Datenbanken*
- „Standardmäßige progressive Typzuordnungen in Unicode - ODBC-Wrapper“ in *Systeme zusammenschlossener Datenbanken*
- „Standardmäßige progressive Typzuordnungen in Unicode - Microsoft SQL Server-Wrapper“ in *Systeme zusammenschlossener Datenbanken*



---

## Anhang I. Standardmäßige regressive Typzuordnungen

Zwischen den Datentypen von Datenquellen und zusammengeschlossenen Datenbanken gibt es zwei Typen von Zuordnungen. Dies ist zum Einen die progressive und zum Anderen die regressive Typzuordnung. In einer *progressiven Typzuordnung* erfolgt die Zuordnung von einem fernen Typ zu einem vergleichbaren lokalen Typ. Der andere Typ der Zuordnung wird als *regressive Typzuordnung* bezeichnet. Dieser Zuordnungstyp wird mit transparenter DDL zur Erstellung oder Änderung von fernen Tabellen verwendet.

Für die meisten Datenquellen sind die Standardtypzuordnungen in den Wrappern enthalten. Die Standardtypzuordnungen für Datenquellen der DB2-Produktfamilie sind im DRDA-Wrapper enthalten. Die Standardtypzuordnungen für Informix sind im Informix-Wrapper enthalten, usw.

Wenn Sie in der zusammengeschlossenen DB2-Datenbank eine ferne Tabelle oder Sicht definieren, schließt die Definition eine regressive Typzuordnung ein. Die Zuordnung erfolgt für jede Spalte von einem *lokalen* Datentyp von DB2 für Linux, UNIX und Windows zu dem entsprechenden *fernen* Datentyp. Es gibt beispielsweise eine standardmäßige regressive Typzuordnung, in der der lokale Datentyp REAL auf den Informix-Datentyp SMALLFLOAT verweist.

Server mit zusammengeschlossenen Datenbanken unter DB2 für Linux, UNIX und Windows unterstützen keine Zuordnungen für benutzerdefinierte Datentypen sowie für die Datentypen LONG VARCHAR, LONG VARGRAPHIC und DATA-LINK.

Bei Verwendung der Anweisung CREATE TABLE zur Erstellung einer fernen Tabelle geben Sie die lokalen Datentypen an, die in der fernen Tabelle vorhanden sein sollen. Die standardmäßigen regressiven Typzuordnungen ordnen diesen Spalten entsprechende ferne Datentypen zu. Angenommen, Sie verwenden die Anweisung CREATE TABLE, um eine Informix-Tabelle mit einer Spalte C2 zu definieren. Sie geben in der Anweisung BIGINT als Datentyp für C2 an. Die standardmäßige regressive Typzuordnung für BIGINT ist abhängig von der Informix-Version, in der die Tabelle erstellt wird. Die Zuordnung für C2 in der Informix-Tabelle lautet für Informix Version 8 DECIMAL und für Informix Version 9 INT8.

Sie können eine standardmäßige regressive Typzuordnung überschreiben oder unter Verwendung der Anweisung CREATE TYPE MAPPING eine neue regressive Typzuordnung erstellen.

In den folgenden Tabellen sind die standardmäßigen regressiven Typzuordnungen zwischen lokalen Datentypen von DB2 für Linux, UNIX und Windows und fernen Datentypen der Datenquellen aufgeführt.

Diese Zuordnungen sind für alle unterstützten Versionen gültig, falls nichts anderes angemerkt ist.

## Datenquellen von DB2 für z/OS und OS/390

Tabelle 165. Standardmäßige regressive Typzuordnungen von DB2 für z/OS und OS/390 (keine vollständige Anzeige der Spalten)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHAR	-	-	N
CHARACTER	-	-	-	-	Y	-	CHAR	-	-	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
I DOUBLE	-	8	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	8	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
I REAL	-	4	-	-	-	-	REAL	-	-	-
I SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	-	Y
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	N

## Datenquellen von DB2 für iSeries

Tabelle 166. Standardmäßige regressive Typzuordnungen von DB2 für iSeries (keine vollständige Anzeige der Spalten)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHARACTER	-	-	N
CHARACTER	-	-	-	-	Y	-	CHARACTER	-	-	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	NUMERIC	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	FLOAT	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	-	Y
VARGRAPHIC	-	-	-	-	-	-	VARG	-	-	N

## Datenquellen von DB2 für VM und VSE

Tabelle 167. Standardmäßige regressive Typzuordnungen von DB2 für VM und VSE (keine vollständige Anzeige der Spalten)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHAR	-	-	-
CHARACTER	-	-	-	-	Y	-	CHAR	-	-	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	REAL	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	-
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	-	Y
VARGRAPH	-	-	-	-	-	-	VARGRAPH	-	-	N

## Datenquellen von DB2 für Linux, UNIX und Windows

Tabelle 168. Standardmäßige regressive Typzuordnungen von DB2 für Linux, UNIX und Windows (keine vollständige Anzeige der Spalten)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	FEDERATED_BIT_DATA
BIGINT	-	8	-	-	-	-	BIGINT	-	-	-
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHARACTER	-	-	-	-	-	-	CHAR	-	-	N
CHARACTER	-	-	-	-	Y	-	CHAR	-	-	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	8	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	N
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
I REAL	-	-	-	-	-	-	REAL	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	10	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	N
VARCHAR	-	-	-	-	Y	-	VARCHAR	-	-	Y
VARGRAPH	-	-	-	-	-	-	VARGRAPHIC	-	-	N
I VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	-



## Informix-Datenquellen

Tabelle 169. Standardmäßige regressive Informix-Datentypzuordnungen

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BIGINT <sup>1</sup>	-	-	-	-	-	-	DECIMAL	19	-	-
BIGINT <sup>2</sup>	-	-	-	-	-	-	INT8	-	-	-
BLOB	1	2147483647	-	-	-	-	BYTE	-	-	-
CHARACTER	-	-	-	-	N	-	CHAR	-	-	-
CHARACTER	-	-	-	-	Y	-	BYTE	-	-	-
CLOB	1	2147483647	-	-	-	-	TEXT	-	-	-
DATE	-	4	-	-	-	-	DATE	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	8	-	-	-	-	FLOAT	-	-	-
INTEGER	-	4	-	-	-	-	INTEGER	-	-	-
REAL	-	4	-	-	-	-	SMALLFLOAT	-	-	-
SMALLINT	-	2	-	-	-	-	SMALLINT	-	-	-
TIME	-	3	-	-	-	-	DATETIME	6	10	-
TIMESTAMP	-	10	-	-	-	-	DATETIME	0	15	-
VARCHAR	1	254	-	-	N	-	VARCHAR	-	-	-
VARCHAR	255	32672	-	-	N	-	TEXT	-	-	-
VARCHAR	-	-	-	-	Y	-	BYTE	-	-	-
VARCHAR <sup>2</sup>	255	2048	-	-	N	-	LVARCHAR	-	-	-
VARCHAR <sup>2</sup>	2049	32672	-	-	N	-	TEXT	-	-	-

### Anmerkungen:

1. Diese Typzuordnung ist nur für Informix Server Version 8 (oder niedriger) zulässig.
2. Diese Typzuordnung ist nur für Informix Server Version 9 zulässig.

Für den Informix-Datentyp DATETIME verwendet der DB2-Server mit zusammengeschlossenen Datenbanken unter UNIX und Windows das Informix-Qualifikationsmerkmal der oberen Ebene als REMOTE\_LENGTH und das Informix-Qualifikationsmerkmal der unteren Ebene als REMOTE\_SCALE.

Die Informix-Qualifikationsmerkmale sind die Konstanten "TU\_", die in der Informix Client SDK-Datei `datatime.h` definiert sind. Es gibt folgende Konstanten:

0 = YEAR	8 = MINUTE	13 = FRACTION(3)
2 = MONTH	10 = SECOND	14 = FRACTION(4)
4 = DAY	11 = FRACTION(1)	15 = FRACTION(5)



Tabelle 170. Standardmäßige regressive Typzuordnungen von Microsoft SQL Server (keine vollständige Anzeige der Spalten) (Forts.)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
VARCHAR	8001	32672	-	-	Y	-	image	-	-	-

**Anmerkungen:**

1. Diese Typzuordnung ist nur für Microsoft SQL Server Version 2000 gültig.

## Oracle NET8-Datenquellen

Tabelle 171. Standardmäßige regressive Oracle NET8-Datentypzuordnungen

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BLOB	0	2147483647	0	0	Y	\0	BLOB	0	0	Y
CHARACTER	1	254	0	0	N	\0	CHAR	0	0	N
CHARACTER	1	254	0	0	Y	\0	RAW	0	0	Y
CLOB	0	2147483647	0	0	N	\0	CLOB	0	0	N
DATE	0	4	0	0	N	\0	DATE	0	0	N
DECIMAL	0	0	0	0	N	\0	NUMBER	0	0	N
DOUBLE	0	8	0	0	N	\0	FLOAT	126	0	N
FLOAT	0	8	0	0	N	\0	FLOAT	126	0	N
INTEGER	0	4	0	0	N	\0	NUMBER	9	0	N
REAL	0	4	0	0	N	\0	FLOAT	63	0	N
SMALLINT	0	2	0	0	N	\0	NUMBER	4	0	N
TIME	0	3	0	0	N	\0	DATE	0	0	N
TIMESTAMP	0	10	0	0	N	\0	DATE	0	0	N

Tabelle 171. Standardmäßige regressive Oracle NET8-Datentypzuordnungen (Forts.)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
VARCHAR	1	4000	0	0	N	\0	VARCHAR2	0	0	N
VARCHAR	1	2000	0	0	Y	\0	RAW	0	0	Y

**Anmerkung:** Der Datentyp BIGINT von DB2 Universal Database für Linux, UNIX und Windows ist für die transparente DDL nicht verfügbar. Beim Erstellen einer fernen Oracle-Tabelle kann in einer Anweisung CREATE TABLE der Datentyp BIGINT nicht angegeben werden.

## Sybase-Datenquellen

Tabelle 172. Standardmäßige regressive Sybase CTLIB-Datentypzuordnungen

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
BIGINT	-	-	-	-	-	-	decimal	19	0	-
BLOB	-	-	-	-	-	-	image	-	-	-
CHARACTER	-	-	-	-	N	-	char	-	-	-
CHARACTER	-	-	-	-	Y	-	binary	-	-	-
CLOB	-	-	-	-	-	-	text	-	-	-
DATE	-	-	-	-	-	-	datetime	-	-	-
DECIMAL	-	-	-	-	-	-	decimal	-	-	-
DOUBLE	-	-	-	-	-	-	float	-	-	-
GRAPHIC	-	-	-	-	-	-	unichar	-	-	-
VARGRAPHIC	-	-	-	-	-	-	univarchar	-	-	-
INTEGER	-	-	-	-	-	-	integer	-	-	-
REAL	-	-	-	-	-	-	real	-	-	-
SMALLINT	-	-	-	-	-	-	smallint	-	-	-

Tabelle 172. Standardmäßige regressive Sybase CTLIB-Datentypzuordnungen (Forts.)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	REMOTE_BIT_DATA
TIME	-	-	-	-	-	-	datetime	-	-	-
TIMESTAMP	-	-	-	-	-	-	datetime	-	-	-
VARCHAR <sup>1</sup>	1	255	-	-	N	-	varchar	-	-	-
VARCHAR <sup>1</sup>	256	32672	-	-	N	-	text	-	-	-
VARCHAR <sup>2</sup>	1	16384	-	-	N	-	varchar	-	-	-
VARCHAR <sup>2</sup>	16385	32672	-	-	N	-	text	-	-	-
VARCHAR <sup>1</sup>	1	255	-	-	Y	-	varbinary	-	-	-
VARCHAR <sup>1</sup>	256	32672	-	-	Y	-	image	-	-	-
VARCHAR <sup>2</sup>	1	16384	-	-	Y	-	varbinary	-	-	-
VARCHAR <sup>2</sup>	16385	32672	-	-	Y	-	image	-	-	-

**Anmerkungen:**

1. Diese Typzuordnung ist nur bei CTLIB mit Sybase Server Version 12.0 (oder einer früheren Version dieses Produkts) zulässig.
2. Diese Typzuordnung ist nur bei CTLIB mit Sybase Server Version 12.5 (oder einer späteren Version dieses Produkts) zulässig.

## Teradata-Datenquellen

Tabelle 173. Standardmäßige regressive Typzuordnungen von Teradata (keine vollständige Anzeige der Spalten)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	FEDERATED_BIT_DATA
BLOB <sup>1</sup>	1	64000	-	-	-	-	VARBYTE	-	-	-
CHARACTER	-	-	-	-	-	-	CHARACTER	-	-	-
CHARACTER	-	-	-	-	Y	-	BYTE	-	-	-

| Tabelle 173. Standardmäßige regressive Typzuordnungen von Teradata (keine vollständige Anzeige der  
| Spalten) (Forts.)

FEDERATED_TYPENAME	FEDERATED_LOWER_LEN	FEDERATED_UPPER_LEN	FEDERATED_LOWER_SCALE	FEDERATED_UPPER_SCALE	FEDERATED_BIT_DATA	FEDERATED_DATA_OPERATORS	REMOTE_TYPENAME	REMOTE_LENGTH	REMOTE_SCALE	FEDERATED_BIT_DATA
CLOB <sup>2</sup>	1	64000	-	-	-	-	VARCHAR	-	-	-
DATE	-	-	-	-	-	-	DATE	-	-	-
DBCLOB <sup>3</sup>	1	32000	-	-	-	-	VARGRAPHIC	-	-	-
DECIMAL	1	18	0	18	-	-	DECIMAL	-	-	-
DECIMAL	19	31	0	31	-	-	FLOAT	-	-	-
DOUBLE	-	-	-	-	-	-	FLOAT	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	-	-
INTEGER	-	-	-	-	-	-	INTEGER	-	-	-
REAL	-	-	-	-	-	-	FLOAT	-	-	-
SMALLINT	-	-	-	-	-	-	SMALLINT	-	-	-
TIME	-	-	-	-	-	-	TIME	-	-	-
TIMESTAMP	-	-	-	-	-	-	TIMESTAMP	-	-	-
VARCHAR	-	-	-	-	-	-	VARCHAR	-	-	-
VARCHAR	-	-	-	-	Y	-	VARBYTE	-	-	-
VARGRAPHIC	-	-	-	-	-	-	VARGRAPHIC	-	-	-

**Anmerkungen:**

1. Der Datentyp VARBYTE von Teradata kann nur die angegebene Länge (1 bis 64000) des DB2-Datentyps BLOB enthalten.
2. Der Datentyp VARCHAR von Teradata kann nur die angegebene Länge (1 bis 64000) des DB2-Datentyps CLOB enthalten.
3. Der Datentyp VARGRAPHIC von Teradata kann nur die angegebene Länge (1 bis 32000) des DB2-Datentyps DBCLOB enthalten.

**Zugehörige Konzepte:**

- „Progressive und regressive Datentypzuordnungen“ in *Systeme zusammengeschlusener Datenbanken*



## Anhang J. Funktionszuordnungsoptionen für Systeme zusammengesessener Datenbanken

DB2 Information Integrator stellt Standardzuordnungen zwischen vorhandenen integrierten Datenquellenfunktionen und den entsprechenden integrierten DB2-Funktionen bereit. Für die meisten Datenquellen sind die standardmäßigen Funktionszuordnungen in den Wrappern enthalten. Zur Verwendung einer Datenquellenfunktion, die der Server mit zusammengesessenen Datenbanken nicht erkennt, muss eine Funktionszuordnung zwischen der Datenquellenfunktion und der entsprechenden Funktion in der zusammengesessenen Datenbank erstellt werden.

Funktionszuordnungsoptionen dienen primär zur Bereitstellung von Informationen über den potenziellen Aufwand für die Ausführung einer Datenquellenfunktion in der Datenquelle. Die Pushdown-Analyse ermittelt, ob mit einer Funktion der Datenquelle eine Funktion in einer Abfrage ausgeführt werden kann. Das Abfrageoptimierungsprogramm entscheidet, ob in der Weiterleitung der Funktionsverarbeitung an die Datenquelle die günstigste Aufwandsalternative besteht.

Die in der Definition der Funktionszuordnung zur Verfügung gestellten statistischen Informationen helfen dem Abfrageoptimierungsprogramm beim Vergleich des geschätzten Aufwands für die Ausführung der Datenquellenfunktion mit dem geschätzten Aufwand für die Ausführung der DB2-Funktion.

*Tabelle 174. Optionen für Funktionszuordnungen und deren Einstellungen*

Option	Gültige Einstellungen	Standardeinstellung
DISABLE	Inaktiviert eine standardmäßige Funktionszuordnung. Gültige Werte sind 'Y' und 'N'.	'N'
INITIAL_INSTS	Geschätzte Anzahl der beim ersten und letzten Aufruf der Datenquellenfunktion verarbeiteten Anweisungen.	'0'
INITIAL_IOS	Geschätzte Anzahl der beim ersten und letzten Aufruf der Datenquellenfunktion ausgeführten Eingaben und Ausgaben.	'0'
IOS_PER_ARGBYTE	Geschätzte Anzahl der für jedes Byte des an die Datenquellenfunktion übergebenen Argumentsatzes aufgewendeten Ein- und Ausgaben.	'0'
IOS_PER_INVOC	Geschätzte Anzahl der Ein- und Ausgaben pro Aufruf einer Datenquellenfunktion.	'0'
INSTS_PER_ARGBYTE	Geschätzte Anzahl der für jedes Byte des an die Datenquellenfunktion übergebenen Argumentsatzes verarbeiteten Anweisungen.	'0'
INSTS_PER_INVOC	Geschätzte Anzahl der bei jedem Aufruf der Datenquellenfunktion verarbeiteten Anweisungen.	'450'
PERCENT_ARGBYTES	Geschätzter durchschnittlicher Prozentsatz der Eingabeargumentbytes, die von der Datenquellenfunktion tatsächlich gelesen werden.	'100'
REMOTE_NAME	Name der Datenquellenfunktion.	lokaler Name





---

## DB2 Information Integrator-Dokumentation

Dieses Thema bietet Informationen zu der für DB2 Information Integrator verfügbaren Dokumentation. In den Tabellen in diesem Thema finden Sie den offiziellen Dokumenttitel, die Formularnummer sowie die Speicherposition der einzelnen PDF-Bücher. Sie benötigen den offiziellen Buchtitel oder die Formularnummer des Dokuments, um ein gedrucktes Handbuch zu bestellen. Die Titel, Dateinamen und Speicherpositionen der Release-Informationen und Installationsvoraussetzungen für DB2 Information Integrator sind in diesem Thema ebenfalls dokumentiert.

Dieses Thema umfasst die folgenden Abschnitte:

- Zugriff auf die DB2 Information Integrator-Dokumentation
- Dokumentation zur Replikationsfunktion unter z/OS
- Dokumentation zur Event-Publishing-Funktion für DB2 Universal Database unter z/OS
- Dokumentation zur Event-Publishing-Funktion für IMS und VSAM unter z/OS
- Dokumentation zur Event-Publishing-Funktion und Replikationsfunktion unter Linux, UNIX und Windows
- Dokumentation zur Funktion für Systeme zusammenschlossener Datenbanken unter z/OS
- Dokumentation zur Funktion für Systeme zusammenschlossener Datenbanken unter Linux, UNIX und Windows
- Dokumentation zu Enterprise Search unter Linux, UNIX und Windows
- Release-Informationen und Installationsvoraussetzungen

---

## Zugriff auf die DB2 Information Integrator-Dokumentation

Alle Handbücher und Release-Informationen zu DB2 Information Integrator sind als PDF-Dateien auf der DB2 Information Integrator-Unterstützungswebsite unter [www.ibm.com/software/data/integration/db2ii/support.html](http://www.ibm.com/software/data/integration/db2ii/support.html) verfügbar.

Sie können über die DB2 Information Integrator-Unterstützungswebsite auf die neueste DB2 Information Integrator-Produktdokumentation zugreifen, indem Sie den Produktinformationslink ('Product Information') anklicken, wie in Abb. 48 auf Seite 710 dargestellt.

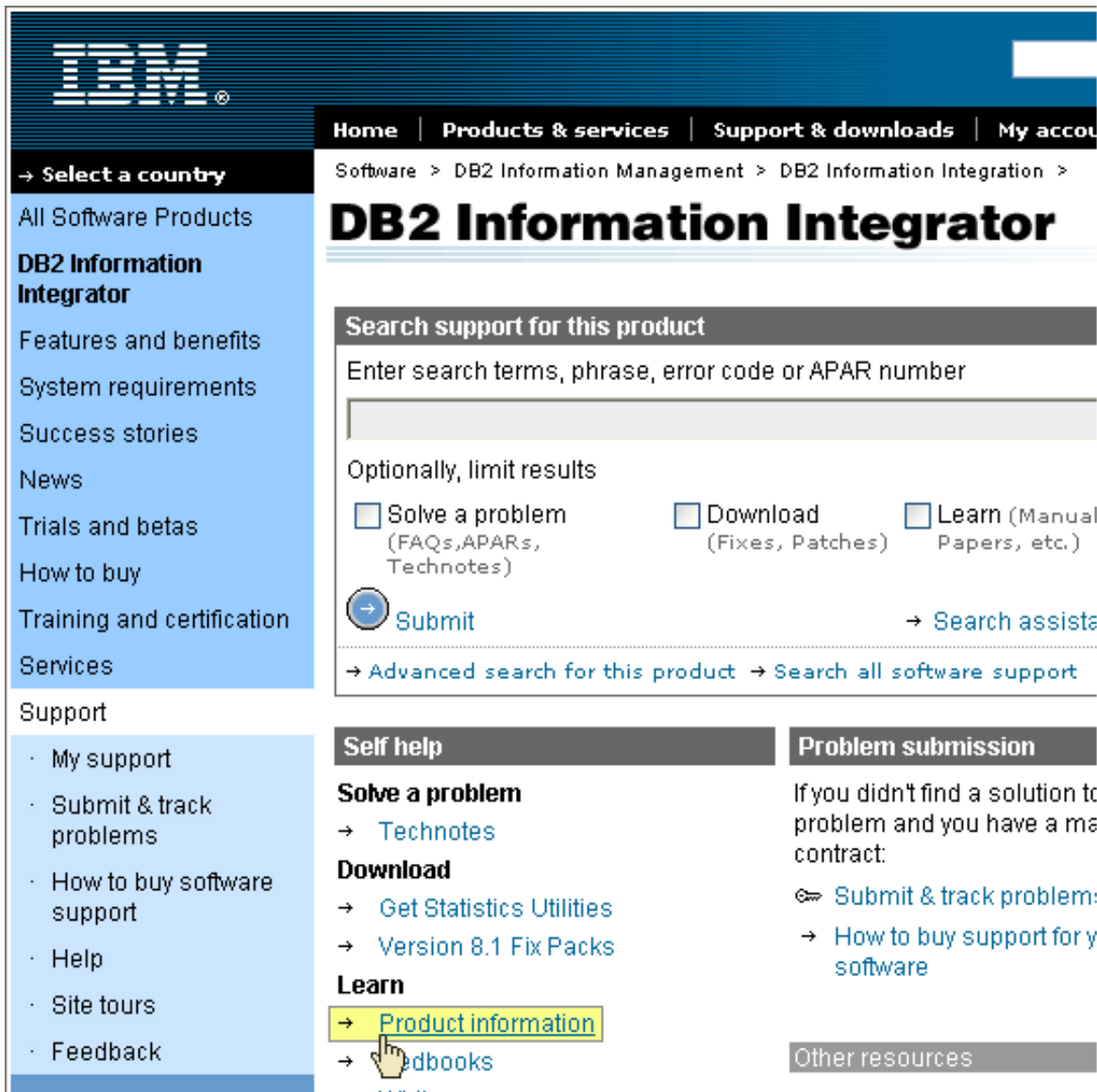


Abbildung 48. Zugriff auf den Produktinformationslink über die DB2 Information Integrator-Unterstützungswebsite

Über den Produktinformationslink besteht Zugriff auf die aktuelle DB2 Information Integrator-Dokumentation in allen unterstützten Sprachen:

- DB2 Information Integrator-Produktdokumentation in PDF-Dateiformat
- Produktdokumentation zu FixPaks, einschließlich Release-Informationen
- Anweisungen zum Herunterladen und Installieren von 'DB2 Information - Unterstützung' für Linux, UNIX und Windows
- Links auf die Onlineversion von 'DB2 Information - Unterstützung'

Blättern Sie in der Liste, um die Produktdokumentation für die von Ihnen verwendete Version von DB2 Information Integrator aufzurufen.

Darüber hinaus stellt die DB2 Information Integrator-Unterstützungswebsite Unterstützungsdokumentation, IBM Redbooks, White Papers, Produktdownloads, Links auf Benutzergruppen sowie Neuerungen zu DB2 Information Integrator zur Verfügung.

Es ist ebenfalls möglich, die PDF-Bücher für DB2 Information Integrator von der CD mit der *DB2-PDF-Dokumentation* aus anzuzeigen und zu drucken.

Gehen Sie wie folgt vor, um die PDF-Dokumentation anzuzeigen oder zu drucken:

1. Öffnen Sie im Stammverzeichnis der CD mit der *DB2-PDF-Dokumentation* die Datei `index.htm`.
2. Klicken Sie die gewünschte Sprache an.
3. Klicken Sie den Link für das Dokument an, das Sie anzeigen möchten.

---

## Dokumentation zur Replikationsfunktion unter z/OS

*Tabelle 175. DB2 Information Integrator-Dokumentation zur Replikationsfunktion unter z/OS*

<b>Name</b>	<b>IBM Form</b>	<b>Speicherposition</b>
<i>ASNCLP Program Reference for Replication and Event Publishing</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Replikation und Event-Publishing - Einführung</i>	GC12-3347	DB2 Information Integrator-Unterstützungswebsite
<i>Migration auf SQL Replication</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"> <li>• CD mit der <i>DB2-PDF-Dokumentation</i></li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Replication Installation and Customization Guide for z/OS</i>	SC18-9127	DB2 Information Integrator-Unterstützungswebsite
<i>SQL Replication Handbuch und Referenz</i>	SC12-3065	<ul style="list-style-type: none"> <li>• CD mit der <i>DB2-PDF-Dokumentation</i></li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Tuning for Replication and Event Publishing Performance</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Tuning for SQL Replication Performance</i>	n/v	DB2 Information Integrator-Unterstützungswebsite

Tabelle 175. DB2 Information Integrator-Dokumentation zur Replikationsfunktion unter z/OS (Forts.)

Name	IBM Form	Speicherposition
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	n/v	<ul style="list-style-type: none"> <li>In 'DB2 Information - Unterstützung': <b>Produktüberblick &gt; Informationsintegration - Überblick &gt; DB2 Information Integrator - Übersicht &gt; Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation</b></li> <li>DB2 Information Integrator - Klickstartleiste für die Installation</li> <li>DB2 Information Integrator-Unterstützungswebsite</li> <li>Produkt-CD von <i>DB2 Information Integrator</i></li> </ul>

## Dokumentation zur Event-Publishing-Funktion für DB2 Universal Database unter z/OS

Tabelle 176. DB2 Information Integrator-Dokumentation zur Event-Publishing-Funktion für DB2 Universal Database unter z/OS

Name	IBM Form	Speicherposition
<i>ASNCLP Program Reference for Replication and Event Publishing</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Replikation und Event-Publishing - Einführung</i>	GC12-3347	<ul style="list-style-type: none"> <li>CD mit der <i>DB2-PDF-Dokumentation</i></li> <li>DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"> <li>CD mit der <i>DB2-PDF-Dokumentation</i></li> <li>DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Tuning for Replication and Event Publishing Performance</i>	n/v	DB2 Information Integrator-Unterstützungswebsite

Tabelle 176. DB2 Information Integrator-Dokumentation zur Event-Publishing-Funktion für DB2 Universal Database unter z/OS (Forts.)

Name	IBM Form	Speicherposition
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	n/v	<ul style="list-style-type: none"> <li>In 'DB2 Information - Unterstützung': <b>Produktüberblick &gt; Informationsintegration - Überblick &gt; DB2 Information Integrator - Übersicht &gt; Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation</b></li> <li>DB2 Information Integrator - Klickstartleiste für die Installation</li> <li>DB2 Information Integrator-Unterstützungswebsite</li> <li>Produkt-CD von <i>DB2 Information Integrator</i></li> </ul>

## Dokumentation zur Event-Publishing-Funktion für IMS und VSAM unter z/OS

Tabelle 177. DB2 Information Integrator-Dokumentation zur Event-Publishing-Funktion für IMS und VSAM unter z/OS

Name	IBM Form	Speicherposition
<i>Client Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9160	DB2 Information Integrator-Unterstützungswebsite
<i>Data Mapper Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9163	DB2 Information Integrator-Unterstützungswebsite
<i>Getting Started with Event Publisher for z/OS</i>	GC18-9186	DB2 Information Integrator-Unterstützungswebsite
<i>Installation Guide for Classic Federation and Event Publisher for z/OS</i>	GC18-9301	DB2 Information Integrator-Unterstützungswebsite
<i>Operations Guide for Event Publisher for z/OS</i>	SC18-9157	DB2 Information Integrator-Unterstützungswebsite
<i>Planning Guide for Event Publisher for z/OS</i>	SC18-9158	DB2 Information Integrator-Unterstützungswebsite
<i>Reference for Classic Federation and Event Publisher for z/OS</i>	SC18-9156	DB2 Information Integrator-Unterstützungswebsite
<i>System Messages for Classic Federation and Event Publisher for z/OS</i>	SC18-9162	DB2 Information Integrator-Unterstützungswebsite
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for IMS for z/OS</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for VSAM for z/OS</i>	n/v	DB2 Information Integrator-Unterstützungswebsite

## Dokumentation zur Event-Publishing-Funktion und Replikationsfunktion unter Linux, UNIX und Windows

Tabelle 178. DB2 Information Integrator-Dokumentation zur Event-Publishing-Funktion und Replikationsfunktion unter Linux, UNIX und Windows

Name	IBM Form	Speicherposition
<i>ASNCLP Program Reference for Replication and Event Publishing</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Installationshandbuch für Linux, UNIX und Windows</i>	GC12-3214	<ul style="list-style-type: none"> <li>• CD mit der DB2-PDF-Dokumentation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Replikation und Event-Publishing - Einführung</i>	GC12-3347	<ul style="list-style-type: none"> <li>• CD mit der DB2-PDF-Dokumentation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Migration auf SQL Replication</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"> <li>• CD mit der DB2-PDF-Dokumentation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>SQL Replication Handbuch und Referenz</i>	SC12-3065	DB2 Information Integrator-Unterstützungswebsite
<i>Tuning for Replication and Event Publishing Performance</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Tuning for SQL Replication Performance</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	n/v	<ul style="list-style-type: none"> <li>• In 'DB2 Information - Unterstützung': <b>Produktüberblick &gt; Informationsintegration - Überblick &gt; DB2 Information Integrator - Übersicht &gt; Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation</b></li> <li>• DB2 Information Integrator - Klickstartleiste für die Installation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> <li>• Produkt-CD von <i>DB2 Information Integrator</i></li> </ul>

## Dokumentation zur Funktion für Systeme zusammengeschlossener Datenbanken unter z/OS

Tabelle 179. DB2 Information Integrator-Dokumentation zur Funktion für Systeme zusammengeschlossener Datenbanken unter z/OS

Name	IBM Form	Speicherposition
<i>Client Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9160	DB2 Information Integrator-Unterstützungswebsite
<i>Data Mapper Guide for Classic Federation and Event Publisher for z/OS</i>	SC18-9163	DB2 Information Integrator-Unterstützungswebsite
<i>Getting Started with Classic Federation for z/OS</i>	GC18-9155	DB2 Information Integrator-Unterstützungswebsite
<i>Installation Guide for Classic Federation and Event Publisher for z/OS</i>	GC18-9301	DB2 Information Integrator-Unterstützungswebsite
<i>Reference for Classic Federation and Event Publisher for z/OS</i>	SC18-9156	DB2 Information Integrator-Unterstützungswebsite
<i>System Messages for Classic Federation and Event Publisher for z/OS</i>	SC18-9162	DB2 Information Integrator-Unterstützungswebsite
<i>Transaction Services Guide for Classic Federation for z/OS</i>	SC18-9161	DB2 Information Integrator-Unterstützungswebsite
<i>Release Notes for IBM DB2 Information Integrator Classic Federation for z/OS</i>	n/v	DB2 Information Integrator-Unterstützungswebsite

## Dokumentation zur Funktion für Systeme zusammengeschlossener Datenbanken unter Linux, UNIX und Windows

Tabelle 180. DB2 Information Integrator-Dokumentation zur Funktion für Systeme zusammengeschlossener Datenbanken unter Linux, UNIX und Windows

Name	IBM Form	Speicherposition
<i>Application Developer's Guide</i>	SC18-7359	<ul style="list-style-type: none"> <li>• CD mit der DB2-PDF-Dokumentation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>C++ API Reference for Developing Wrappers</i>	SC18-9172	<ul style="list-style-type: none"> <li>• CD mit der DB2-PDF-Dokumentation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Konfiguration von Datenquellen</i>	n/v	<ul style="list-style-type: none"> <li>• CD mit der DB2-PDF-Dokumentation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Systeme zusammengeschlossener Datenbanken</i>	SC12-3241	<ul style="list-style-type: none"> <li>• CD mit der DB2-PDF-Dokumentation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Guide to Configuring the Content Connector for VeniceBridge</i>	n/v	DB2 Information Integrator-Unterstützungswebsite



Tabelle 180. DB2 Information Integrator-Dokumentation zur Funktion für Systeme zusammengeglichener Datenbanken unter Linux, UNIX und Windows (Forts.)

Name	IBM Form	Speicherposition
<i>Installationshandbuch für Linux, UNIX und Windows</i>	GC12-3214	<ul style="list-style-type: none"> <li>• CD mit der <i>DB2-PDF-Dokumentation</i></li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Java API Reference for Developing Wrappers</i>	SC18-9173	<ul style="list-style-type: none"> <li>• CD mit der <i>DB2-PDF-Dokumentation</i></li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Migration</i>	SC12-3243	<ul style="list-style-type: none"> <li>• CD mit der <i>DB2-PDF-Dokumentation</i></li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Wrapper Developer's Guide</i>	SC18-9174	<ul style="list-style-type: none"> <li>• CD mit der <i>DB2-PDF-Dokumentation</i></li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> </ul>
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	n/v	<ul style="list-style-type: none"> <li>• In 'DB2 Information - Unterstützung': <b>Produktüberblick &gt; Informationsintegration - Überblick &gt; DB2 Information Integrator - Übersicht &gt; Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation</b></li> <li>• DB2 Information Integrator - Klickstartleiste für die Installation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> <li>• Produkt-CD von <i>DB2 Information Integrator</i></li> </ul>

## Dokumentation zu Enterprise Search unter Linux, UNIX und Windows

Tabelle 181. DB2 Information Integrator-Dokumentation zu Enterprise Search unter Linux, UNIX und Windows

Name	IBM Form	Speicherposition
<i>Administering Enterprise Search</i>	SC18-9283	DB2 Information Integrator-Unterstützungswebsite
<i>Installation Guide for Enterprise Search</i>	GC18-9282	DB2 Information Integrator-Unterstützungswebsite
<i>Programming Guide and API Reference for Enterprise Search</i>	SC18-9284	DB2 Information Integrator-Unterstützungswebsite
<i>Release Notes for Enterprise Search</i>	n/v	DB2 Information Integrator-Unterstützungswebsite

## Release-Informationen und Installationsvoraussetzungen

Release-Informationen enthalten die für die jeweilige Release- und FixPak-Stufe des verwendeten Produkts spezifischen Informationen sowie die neuesten Dokumentationskorrekturen für das jeweilige Release.

Die Installationsvoraussetzungen enthalten Informationen, die für das jeweilige Release des verwendeten Produkts spezifisch sind.

*Tabelle 182. DB2 Information Integrator Release-Informationen und Installationsvoraussetzungen*

<b>Name</b>	<b>Dateiname</b>	<b>Speicherposition</b>
<i>Installation Requirements for IBM DB2 Information Integrator Event Publishing Edition, Replication Edition, Standard Edition, Advanced Edition, Advanced Edition Unlimited, Developer Edition, and Replication for z/OS</i>	Prereqs	<ul style="list-style-type: none"> <li>• Produkt-CD von <i>DB2 Information Integrator</i></li> <li>• DB2 Information Integrator - Klickstartleiste für die Installation</li> </ul>
<i>Release Notes for IBM DB2 Information Integrator Standard Edition, Advanced Edition, and Replication for z/OS</i>	ReleaseNotes	<ul style="list-style-type: none"> <li>• In 'DB2 Information - Unterstützung': <b>Produktüberblick &gt; Informationsintegration - Überblick &gt; DB2 Information Integrator - Übersicht &gt; Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation</b></li> <li>• DB2 Information Integrator - Klickstartleiste für die Installation</li> <li>• DB2 Information Integrator-Unterstützungswebsite</li> <li>• Produkt-CD von <i>DB2 Information Integrator</i></li> </ul>
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for IMS for z/OS</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Release Notes for IBM DB2 Information Integrator Event Publisher for VSAM for z/OS</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Release Notes for IBM DB2 Information Integrator Classic Federation for z/OS</i>	n/v	DB2 Information Integrator-Unterstützungswebsite
<i>Release Notes for Enterprise Search</i>	n/v	DB2 Information Integrator-Unterstützungswebsite

Gehen Sie wie folgt vor, um die Installationsvoraussetzungen und Release-Informationen auf der Produkt-CD anzuzeigen:

- Geben Sie unter Windows Folgendes ein:

`x:\doc\%L`

Dabei ist *x* der Laufwerksbuchstabe des CD-Laufwerks unter Windows, und *%L* ist die Ländereinstellung der zu verwendenden Dokumentation, z. B. `de_DE`.

- Geben Sie unter UNIX Folgendes ein:

`/cdrom/doc/%L/`

Dabei ist *cdrom* der Mountpunkt der CD unter UNIX, und *%L* ist die Ländereinstellung der zu verwendenden Dokumentation, z. B. `de_DE`.

---

## Eingabehilfen

Eingabehilfen unterstützen Benutzer mit körperlichen Behinderungen, wie z. B. eingeschränkter Bewegungsfähigkeit oder Sehkraft, beim erfolgreichen Einsatz von Softwareprodukten. Im Folgenden sind die wichtigsten Eingabehilfen aufgeführt, die in den Produkten von DB2<sup>®</sup> Version 8 zur Verfügung stehen:

- Die gesamte DB2-Funktionalität kann sowohl über die Maus als auch über die Tastatur gesteuert werden. Weitere Informationen hierzu finden Sie unter „Tastatureingabe und Navigation“.
- Sie können die Größe und Farbe der verwendeten Schriftarten in den DB2-Schnittstellen anpassen. Weitere Informationen hierzu finden Sie unter „Eingabehilfen für Bildschirme“ auf Seite 720.
- DB2-Produkte unterstützen Anwendungen mit Eingabehilfen, die mit der Java<sup>™</sup> Accessibility API arbeiten. Weitere Informationen hierzu finden Sie unter „Kompatibilität mit Unterstützungseinrichtungen“ auf Seite 720.
- Die DB2-Dokumentation steht in behindertengerechtem Format zur Verfügung. Weitere Informationen hierzu finden Sie unter „Dokumentation im behindertengerechten Format“ auf Seite 720.

---

## Tastatureingabe und Navigation

### Tastatureingabe

Die verfügbaren DB2-Tools können unter ausschließlicher Benutzung der Tastatur verwendet werden. Mit entsprechenden Tasten oder Tastenkombinationen können Operationen ausgeführt werden, die auch über die Maus verfügbar sind. Die Standardtastenkombinationen des Betriebssystems werden für die entsprechenden Standardoperationen des Betriebssystems verwendet.

Weitere Informationen zur Verwendung von Tasten oder Tastenkombinationen für die Ausführung von Operationen finden Sie unter " 'Direktaufrufe über die Tastatur: Gemeinsame GUI - Hilfe'.

### Navigation über die Tastatureingabe

Sie können in den Benutzerschnittstellen der DB2-Tools mit Hilfe von Tasten oder Tastenkombinationen navigieren.

Weitere Informationen zur Navigation in den DB2-Tools mit Hilfe der Tastatureingabe finden Sie unter " 'Direktaufrufe über die Tastatur: Gemeinsame GUI - Hilfe'.

### Tastatureingabebereich

Unter UNIX<sup>®</sup>-Betriebssystemen ist der Bereich des aktiven Fensters, in dem die Tastatureingabe wirksam ist, hervorgehoben.

---

## Eingabehilfen für Bildschirme

Die DB2-Tools stellen Funktionen bereit, mit denen sehbehinderten Benutzern verbesserten Eingabehilfen zur Verfügung stehen. Diese Eingabehilfen umfassen die Unterstützung individuell anpassbarer Schriftarteigenschaften.

### Schriftarteinstellungen

Über das Notizbuch 'Tools - Einstellungen' können Sie die Farbe, Größe und Schriftart des Textes in Menüs und Dialogfenstern auswählen.

Weitere Informationen zur Angabe von Schriftarteinstellungen finden Sie unter "Ändern der Schriftarten für Menüs und Text: Gemeinsame GUI - Hilfe".

### Unabhängigkeit von Farben

Zur Verwendung der Funktionen des vorliegenden Produkts ist es nicht erforderlich, zwischen unterschiedlichen Farben differenzieren zu können.

---

## Kompatibilität mit Unterstützungseinrichtungen

Die Schnittstellen der DB2-Tools unterstützen die Java Accessibility API. Hierdurch wird der Einsatz von Sprachausgabeprogrammen und anderen Unterstützungseinrichtungen für Personen mit Behinderungen mit den DB2-Produkten ermöglicht.

---

## Dokumentation im behindertengerechten Format

Die Dokumentation für DB2 steht im Format XHTML 1.0 zur Verfügung, das mit den meisten Webbrowsern geöffnet werden kann. XHTML ermöglicht das Aufrufen der Dokumentation mit den Anzeigeeinstellungen, die Sie in Ihrem Browser definiert haben. Darüber hinaus ist der Einsatz von Sprachausgabeprogrammen und anderen Unterstützungseinrichtungen möglich.

Syntaxdiagramme stehen in der Schreibweise mit Trennzeichen zur Verfügung. Dieses Format ist nur dann verfügbar, wenn Sie mit Hilfe eines Sprachausgabeprogramms auf die Onlinedokumentation zugreifen.

#### Zugehörige Konzepte:

- „Syntaxdiagramme in der Schreibweise mit Trennzeichen“ in den *Themen zur Infrastruktur (DB2 - Allgemeine Informationen)*

#### Zugehörige Tasks:

- „Direktaufrufe über die Tastatur: Gemeinsame GUI - Hilfe“
- „Ändern der Schriftarten für Menüs und Text: Gemeinsame GUI - Hilfe“

---

## 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. An Stelle 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 der Produkte, Programme oder Dienstleistungen in Verbindung mit Fremdprodukten und Fremddienstleistungen liegt beim Kunden, soweit solche Verbindungen nicht ausdrücklich von IBM bestätigt sind.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Europe, Director of Licensing, 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 bekanntgegeben. IBM kann 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  
J46A/G4  
555 Bailey Avenue  
San Jose, CA 95141-1003  
U.S.A.

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Handbuch aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der Allgemeinen Geschäftsbedingungen der IBM (ICA), der Internationalen Nutzungsbedingungen der IBM für Programmpakete (IPLA) 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 Garantie, 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.

Informationen über Produkte anderer Hersteller als IBM wurden von den Herstellern dieser Produkte zur Verfügung gestellt, bzw. aus von ihnen veröffentlichten Ankündigungen oder anderen öffentlich zugänglichen Quellen entnommen. IBM hat diese Produkte nicht getestet und übernimmt im Hinblick auf Produkte anderer Hersteller keine Verantwortung für einwandfreie Funktion, Kompatibilität oder andere Ansprüche. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten der IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele der IBM.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogrammes illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden, Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

#### COPYRIGHT-LIZENZ:

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. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten.

Kopien oder Teile der Beispielprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (Name Ihrer Firma) (Jahr). Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corp. abgeleitet. © Copyright IBM Corp. *Jahr/Jahre angeben*. Alle Rechte vorbehalten.

---

## Marken

Folgende Namen sind in gewissen Ländern Marken der International Business Machines Corporation:

IBM  
DB2  
DB2 Universal  
Database  
Domino  
Domino.Doc  
Informix  
Lotus  
Lotus Discovery Server  
Lotus Notes  
QuickPlace  
Sametime  
SecureWay  
WebSphere

Folgende Namen sind in gewissen Ländern Marken oder eingetragene Marken anderer Unternehmen:

Java und alle auf Java basierenden Marken und Logos sind in gewissen Ländern Marken oder eingetragene Marken der Sun Microsystems, Inc.

Microsoft, Windows, Windows NT und das Windows-Logo sind in gewissen Ländern Marken der Microsoft Corporation.

Intel, Intel Inside (Logos), MMX und Pentium sind in gewissen Ländern Marken der Intel Corporation.

UNIX ist in gewissen Ländern eine eingetragene Marke von The Open Group.

Andere Namen von Unternehmen, Produkten oder Dienstleistungen können Marken oder Dienstleistungsmarken anderer Unternehmen sein.





---

# Index

## A

Abfragen  
auf Datenquellen zugreifen  
  BioRS 90, 93  
  BLAST 131, 132  
  Documentum 212, 214  
  Entrez 232, 233  
  Excel 254  
  Extended Search 278  
  Geschäftsanwendungen 175  
  HMMER 309  
  Web-Services 463  
  XML 489  
Aufwandsmodell für XML-Dateien 481  
Beispiele  
  Geschäftsanwendungen 172, 459  
  Web-Services-Wrapper 172, 459  
Fragmente 10  
Abfrageoptimierung  
  Beschreibung 10  
  Richtlinien für BioRS-Abfragen 98  
  XML 488  
ACCOUNTING\_STRING, Benutzeroption gültige Einstellungen 659  
Aktionsausgabe, Dienstprogramm  
  Server mit zusammengeschlossenen Datenbanken konfigurieren 66  
Aktualisieren  
  Statistikdaten planen 28  
Allgemeine Abfragesprache  
  Extended Search 281  
AllNbrsByGeneS, benutzerdefinierte Funktion  
  Syntax 561  
AllNbrsByGeneT, benutzerdefinierte Funktion  
  Syntax 562  
AllText, Element  
  BioRS 92  
ALTER NICKNAME, Anweisung  
  Beispiel  
    lokale Spaltennamen ändern 590  
    lokaler Datentyp 597  
    Spaltenoptionen ändern 593  
  Einschränkungen 589  
Ändern  
  Kurznamen  
    Kurznamenoptionen 592  
    lokaler Datentyp 595  
    Übersicht 587  
Anwendungsdatenentität  
  Geschäftsobjektdefinition 139

## B

Beachtung der Groß-/Kleinschreibung  
  Beibehalten von Werten, bei denen die Groß-/Kleinschreibung beachtet werden muss 26  
  Prüfliste für zusammengeschlossene Datenquellen 36  
Befehlszeilenprozessor (CLP)  
  Funktionen zusammengeschlossener Datenbanken 22  
Befehlszentrale  
  Datenquellen konfigurieren 63  
  für Systeme zusammengeschlossener Datenbanken verwenden 22  
Beispiele  
  Abfragen  
    Extended Search 276  
  Codon-Frequenztafel 542  
  db2blast\_daemon, Befehl 120  
  db2hmmr\_daemon, Befehl 297  
  Excel-Dateien 30, 249  
  Geschäftsanwendungswrapper 172, 459  
  Web-Services-Wrapper 172, 451, 459  
  WebSphere Business Integration-Wrapper 158  
  XML-Dokumentzuordnung 471  
Beispiele für Abfragen  
  BioRS 90, 93  
  BLAST 132  
  Dateien mit Tabellenstruktur 404  
  Documentum 212  
  Entrez 233  
  Excel 254  
  Extended Search 278  
  Geschäftsanwendungsdatenquellen 175  
  HMMER 308  
  Web-Services 463  
  XML 489  
Beispiele für Anweisung CREATE USER MAPPING  
  Datenquellen der DB2-Produktfamilie 187  
  Informix 324  
  Microsoft SQL Server 341  
  ODBC 355  
  OLE DB 370  
  Oracle 381  
  Sybase 398  
  Teradata 425  
Beispiele für Anweisung CREATE WRAPPER  
  ODBC 351  
Beispiele für benutzerdefinierte Funktionen (UDFs)  
  LSBarCode, benutzerdefinierte Funktion 528  
  LSDefineParse, benutzerdefinierte Funktion 520

Beispiele für benutzerdefinierte Funktionen (UDFs) (Forts.)  
  LSMultiMatch, benutzerdefinierte Funktion 530  
  LSMultiMatch3, benutzerdefinierte Funktion 531  
  LSNuc2Pep, benutzerdefinierte Funktion 538  
  LSPatternMatch, benutzerdefinierte Funktion 521  
  LSPep2AmbNuc, benutzerdefinierte Funktion 510  
  LSPep2ProbNuc, benutzerdefinierte Funktion 512  
  LSPrositePattern, benutzerdefinierte Funktion 524  
  LSRevComp, benutzerdefinierte Funktion 534  
  LSRevNuc, benutzerdefinierte Funktion 535  
  LSRevPep, benutzerdefinierte Funktion 536  
  LSTransAllFrames 540  
  Übersetzungstabelle für Life Sciences 543  
Beispiele für Dämon  
  BLAST-Konfigurationsdatei 118  
  HMMER-Konfigurationsdatei 295  
Benutzer-IDs  
  Werte, bei denen die Groß-/Kleinschreibung beachtet werden muss 26  
Benutzer mit körperlichen Behinderungen 719  
Benutzerdefinierte Funktionen (UDFs) 20  
  entfernen (KEGG) 577  
  entfernen (Life Sciences) 507  
  KEGG 545, 546  
  Life Sciences 503, 504  
  registrieren (KEGG) 548  
  registrieren (Life Sciences) 505  
Benutzerdefinierte Life Sciences-Funktionen  
  Beispiel für Übersetzungstabelle 543  
  entfernen 507  
  Format von Übersetzungstabellen 542  
  GeneWise 525  
  Liste der Funktionen 504  
  registrieren 505  
  reguläre Ausdrücke, Unterstützung 524  
  Übersicht 503  
  Übersicht über benutzerdefinierte Funktionen zur Syntexanalyse von Definitionszeilen 514  
  Übersicht über Rückübersetzungsfunktionen 508

- Benutzerdefinierte Typen (User-Defined Types, UDTs)
    - nicht unterstützte Datentypen 20
  - Benutzerzuordnungen
    - Beschreibung 16
    - gültige Einstellungen 659
    - Optionen 16
    - planen 34
    - Prüfliste für zusammengeschlossene Datenquellen 36
    - Übersicht zur Registrierung 72
    - Web-Services-Wrapper 467
  - Berechtigungen
    - Dateizugriff für Dateien mit Tabellenstruktur 410
    - Dateizugriff für Excel-Dateien 256
  - BestHmlgsByGenesS, benutzerdefinierte Funktion
    - Syntax 572
  - BestHmlgsByGenesT, benutzerdefinierte Funktion
    - Syntax 573
  - BestNbrsByGenesS, benutzerdefinierte Funktion
    - Syntax 566
  - BestNbrsByGeneT, benutzerdefinierte Funktion
    - Syntax 567
  - Bibliotheken
    - benutzerdefinierte Funktionen 503
    - Informix-Wrapper 320
    - manuelles Herstellen der Programmverbindung zu den Clientbibliotheken 47
    - Microsoft SQL Server-Wrapper 337
    - Oracle-Wrapper 378
    - Sybase-Wrapper 393
    - Teradata-Wrapper 422
    - Wrapper 44, 45
  - Biomax Informatics 77
  - BioRS
    - AllText, Element 92
    - angepasste Funktionen registrieren 81
    - Beispielabfragen 90, 93
    - Beschreibung 77
    - CREATE NICKNAME, Anweisung (Beispiel) 86
    - CREATE NICKNAME, Syntax 601
    - CREATE SERVER, Anweisung 599
    - CREATE USER MAPPING, Anweisung 83, 600
    - kundenspezifische Funktionen 80, 88
    - Nachrichten 104
    - Registrieren von Kurznamen 84
    - Serverdefinitionen registrieren 83
    - Statistikdaten
      - Datenbankkardinalität 100
      - Kurznamenkardinalität 100
      - Spaltenkardinalität 101, 103
      - verwalten 99
    - unterstützte Datentypen 581
    - Verknüpfungsprädikate verwenden 90
    - Wrapper registrieren 81
    - Wrapperbibliotheksdateien 82
    - Wrapperleistung optimieren 98
  - BioRS (*Forts.*)
    - zu einem System zusammengeschlossener Datenbanken hinzufügen 79
    - Zugriff konfigurieren 79
  - BLAST
    - Abfragen 131
    - Beispielabfragen 132
    - CREATE NICKNAME, Anweisung 123
    - CREATE NICKNAME, Anweisung (Beispiel) 130
    - CREATE NICKNAME, Syntax 603
    - CREATE SERVER, Anweisung 602
    - Dämon
      - konfigurieren 115
      - starten 119
    - Dämonkonfigurationsdatei
      - Beispiele 118
    - Definitionszeilen, Syntaxanalyse 124
    - fixierte Eingabe- und Ausgabespalten 124
    - gültige Objekte für Kurznamen 18
    - High-Scoring Segment Pairs (HSP) 109
    - Konfiguration optimieren 134
    - Nachrichten 134
    - Registrieren von Kurznamen 123
    - Serverdefinitionen registrieren 122
    - TurboBlast konfigurieren 130
    - unterstützte Datentypen 581
    - unterstützte Versionen 6
    - Wrapper registrieren 121
    - Wrapperbibliotheksdateien 122
    - zu einem System zusammengeschlossener Datenbanken hinzufügen 113
      - Überprüfen, dass die korrekte ausführbare Datei 'blastall' installiert ist 114
      - Überprüfen, dass die korrekten Matrixdateien installiert sind 114
      - Zugriff konfigurieren 113
  - BLAST\_OUT\_DIR\_PATH
    - BLAST-Dämon 115
  - BLASTALL\_PATH
    - BLAST-Dämon 115
  - BstBstHmlgByGenesS, benutzerdefinierte Funktion
    - Syntax 573
  - BstBstHmlgByGenesT, benutzerdefinierte Funktion
    - Syntax 574
  - BstBstNbrsByGeneS, benutzerdefinierte Funktion
    - Syntax 564
  - BstBstNbrsByGeneT, benutzerdefinierte Funktion
    - Syntax 565
- C**
- CATALOG DATABASE, Befehl
    - auf Datenquellen der DB2-Produktfamilie zugreifen 181
  - CATALOG TCP/IP NODE, Befehl
    - auf Datenquellen der DB2-Produktfamilie zugreifen 180
  - CC-Tool (Connector Configurator = Konnektorkonfigurationsprogramm)
    - Konfiguration von Geschäftsobjektanwendungen 141
  - Clientbibliotheken
    - Documentum 194
  - CLP (Befehlszeilenprozessor)
    - Funktionen zusammengeschlossener Datenbanken 22
  - Codepages 56
    - Systeme zusammengeschlossener Datenbanken 52
  - Codon-Frequenztabelle 541, 542
  - COLLATING\_SEQUENCE, Serveroption
    - gültige Einstellungen 641
  - COMM\_RATE, Serveroption
    - gültige Einstellungen 641
  - CompoundsByPathwyS, benutzerdefinierte Funktion
    - Syntax 552
  - CompoundsByPathwyT, benutzerdefinierte Funktion
    - Syntax 552
  - CONNECTSTRING, Serveroption
    - gültige Einstellungen 641
  - CPU\_RATIO, Serveroption
    - gültige Einstellungen 641
  - CREATE FUNCTION, Anweisung
    - Extended Search 275, 616
  - CREATE INDEX, Anweisung 21
  - CREATE NICKNAME, Anweisung 75
    - BioRS 84, 86
    - BLAST 123, 130
    - Dateien mit Tabellenstruktur 408, 409
    - Datenquellen der DB2-Produktfamilie 189, 190
    - Documentum 199, 204
    - Entrez 228, 617
    - Excel-Dateien 253
    - Extended Search 274
    - Geschäftsanwendungsdatenquellen 148
    - HMMER 302, 307
    - Informix 326, 327
    - Microsoft SQL Server 344, 345
    - ODBC 357, 358
    - Oracle 384, 385
    - Sybase 400, 401
    - Teradata 428, 429
    - Web-Services 442
    - WebSphere Business Integration-Wrapper 148
    - XML 482
  - CREATE NICKNAME, Syntax
    - BioRS 601
    - BLAST 603
    - Dateien mit Tabellenstruktur 618
    - Documentum 606
    - Excel 609
    - Extended Search 613
    - XML 623
  - CREATE SERVER, Anweisung 4
    - BioRS 83, 599
    - BLAST 122, 602
    - Dateien mit Tabellenstruktur 408
    - Datenquellen der DB2-Produktfamilie 184, 185

CREATE SERVER, Anweisung (Forts.)  
 Documentum 198, 604  
 Entrez 226, 617  
 Excel 609  
 Excel-Dateien 253  
 Extended Search 272, 611  
 HMMER 300  
 Informix 321, 322  
 Microsoft SQL Server 338, 339  
 ODBC 352, 353  
 OLE DB 367, 368  
 Oracle 378, 379  
 Sybase 394, 395  
 Teradata 422, 424  
 Web-Services 441  
 XML 477

CREATE USER MAPPING, Anweisung  
 BioRS 83, 600  
 Datenquellen der DB2-Produktfamilie 186, 187  
 Documentum 198, 605  
 Extended Search 273, 613  
 Informix 323, 324  
 Microsoft SQL Server 341  
 ODBC 354, 355  
 OLE DB 369, 370  
 Oracle 381  
 Sybase 397, 398  
 Teradata 425

CREATE WRAPPER, Anweisung  
 Extended Search 611  
 ODBC 351

## D

DAEMON\_LOGFILE\_DIR  
 BLAST-Dämon 115  
 HMMER-Dämon 291

DAEMON\_PORT  
 BLAST-Dämon 115  
 HMMER-Dämon 291

Dämon  
 für BLAST konfigurieren 115  
 für BLAST starten 119  
 für HMMER konfigurieren 291  
 für HMMER starten 296  
 Konfigurationsdatei  
 BLAST, Beispiele 118  
 HMMER, Beispiele 295

DATALINK, Datentyp  
 nicht unterstützt 20

DATEFORMAT, Serveroption  
 gültige Einstellungen 641

Dateien mit Tabellenstruktur  
 Beispiel 403  
 Beispielabfragen 404  
 CREATE NICKNAME, Anweisung (Beispiel) 409  
 CREATE NICKNAME, Syntax 618  
 gültige Objekte für Kurznamen 18  
 Modell für die Dateizugriffsteuerung 410  
 Nachrichten 411  
 Optimierung 411  
 Registrieren von Kurznamen 408  
 Serverdefinitionen registrieren 408  
 Typen 404

Dateien mit Tabellenstruktur (Forts.)  
 Übersicht 403  
 unterstützte Datentypen 581  
 unterstützte Versionen 6  
 Wrapper registrieren 407  
 Wrapperbibliotheksdateien 407  
 zu einem System zusammengeschlossener Datenbanken hinzufügen 406  
 Zugriff konfigurieren 406  
 Zugriff mit DB2 Information Integrator 404

Dateitypen  
 Dateien mit Tabellenstruktur 404

Datenbank  
 Beschreibung 77

Datenelemente  
 Umsetzung für Entrez 232

Datenquellen 8, 10  
 Beschreibung 5  
 Geschäftsanwendungen 144  
 gültige Servertypen 635  
 konfigurieren 63  
 optionale Konfigurationsschritte 76  
 Standardwrappernamen 14

Datenquellen der DB2-Produktfamilie  
 Benutzerzuordnungen registrieren 186  
 CREATE NICKNAME, Anweisung (Beispiel) 190  
 CREATE SERVER, Anweisung 185  
 CREATE USER MAPPING, Anweisung 187  
 Katalogisieren von fernen Datenbanken 181  
 Katalogisieren von Knoteneinträgen 180  
 Registrieren von Kurznamen 189  
 Serverdefinitionen registrieren 184  
 Serververbindungen testen 188  
 Wrapper registrieren 182  
 Wrapperbibliotheksdateien 183  
 zu einem System zusammengeschlossener Datenbanken hinzufügen 179  
 Zugriff konfigurieren 179

Datenquellen konfigurieren  
 Kurznamenoptionen 661  
 über DB2-Steuerzentrale 64

Datenquellenobjekte  
 Beschreibung 17  
 gültige Objekttypen 18

Datentypen  
 für nicht relationale Datenquellen 581  
 nicht unterstützt 20

Datentypzuordnungen  
 Beschreibung 20  
 degressive 695  
 für ein bestimmtes Datenquellenobjekt 595, 597  
 planen 34  
 progressive 679  
 Prüfliste für zusammengeschlossene Datenquellen 36

DB2\_DJ\_COMM, Umgebungsvariable  
 Informix, Optimierung 328  
 Microsoft SQL Server, Optimierung 345

DB2\_DJ\_COMM, Umgebungsvariable (Forts.)  
 ODBC, Optimierung 363  
 Oracle, Optimierung 386  
 Sybase, Optimierung 402  
 Teradata, Optimierung 430

DB2 für iSeries  
 gültige Objekte für Kurznamen 18  
 gültige Servertypen 635  
 standardmäßige progressive Typzuordnungen 679  
 standardmäßige regressive Typzuordnungen 695  
 Standardwrappername 14  
 unterstützte Versionen 6

DB2 für Linux, UNIX und Windows  
 gültige Objekte für Kurznamen 18  
 gültige Servertypen 635  
 standardmäßige progressive Typzuordnungen 679  
 standardmäßige regressive Typzuordnungen 695  
 Standardwrappername 14  
 unterstützte Versionen 6

DB2 für VM und VSE  
 gültige Objekte für Kurznamen 18  
 gültige Servertypen 635  
 standardmäßige progressive Typzuordnungen 679  
 standardmäßige regressive Typzuordnungen 695  
 Standardwrappername 14  
 unterstützte Versionen 6

DB2 für z/OS und OS/390  
 gültige Objekte für Kurznamen 18  
 gültige Servertypen 635  
 standardmäßige progressive Typzuordnungen 679  
 standardmäßige regressive Typzuordnungen 695  
 Standardwrappername 14  
 unterstützte Versionen 6

DB2\_MAXIMAL\_PUSHDOWN, Serveroption  
 gültige Einstellungen 641

db2blast\_daemon, Befehl 120  
 absetzen (erforderliche Schritte) 119

db2dj.ini  
 Einschränkungen 68  
 Konfiguration von Exemplaren mit mehreren Partitionen 69

db2hmmmer\_daemon, Befehl  
 absetzen (erforderliche Schritte) 296  
 Beispiele 297

db2nodes.cfg, Datei  
 zusammengeschlossene Datenbank 58

DBNAME, Serveroption  
 gültige Einstellungen 641

DefinitionsByGeneS, benutzerdefinierte Funktion  
 Syntax 576

Definitionszeilen, Syntaxanalyse  
 Beschreibung 124

Definitionszeilen, Syntaxanalyse (benutzerdefinierte Funktionen, UDFs)  
 Beschreibung 124

- Definitionszeilen, Syntaxanalyse (benutzerdefinierte Funktionen, UDFs) (*Forts.*)
  - LSDefineParse 514
  - Übersicht 514
- Degressive Typzuordnungen
  - Standardzuordnungen 695
- Direktaufrufe über die Tastatur
  - Unterstützung 719
- DISABLE, Funktionszuordnungsoption
  - gültige Einstellungen 707
- djxlink
  - Fehlernachrichtendatei 46
  - manuelles Herstellen der Programmverbindung zu den Clientbibliotheken 47
  - Prüfen der Programmverbindung zu den Clientbibliotheken 43
- dmcl.ini, Datei
  - Documentum 195
- Documentum
  - Beispiel 191
  - Beispielabfragen 212
  - Benutzerzugriff auf Dokumente 214
  - Benutzerzuordnungen registrieren 198
  - Beschreibung 191
  - Clientbibliotheken 194
  - CREATE NICKNAME, Anweisung 199
  - CREATE NICKNAME, Anweisung (Beispiel) 204
  - CREATE NICKNAME, Syntax 606
  - CREATE SERVER, Anweisung 604
  - CREATE USER MAPPING, Anweisung 198, 605
  - doppelte Definition von
    - Wiederholungsattributen 205
  - gültige Objekte für Kurznamen 18
  - kundenspezifische Funktionen 206
  - Kurznamenspalten 205
  - Nachrichten 214
  - Pseudospalten 200
  - Registrieren von Kurznamen 199
  - Serverdefinitionen registrieren 198
  - Umgebungsvariablen einrichten 195
  - unterstützte Datentypen 581
  - unterstützte Versionen 6
  - Wrapper registrieren 196
  - Wrapperbibliotheksdateien 197
  - zu einem System zusammengeschlusener Datenbanken hinzufügen 193
  - Zugriff konfigurieren 193
- Durchgriffsfunktion
  - Beschreibung 12
  - Einschränkungen 12

## E

- Einschränkungen
  - db2dj.ini, Datei 68
  - Kurznamen ändern 589
- Entrez
  - angepasste Funktionen registrieren 224
  - Beispielabfragen 233
  - Beschreibung 221

- Entrez (*Forts.*)
  - CREATE SERVER, Anweisung 617
  - Datenelemente in SQL-Schemata 232
  - gültige Objekte für Kurznamen 18
  - kundenspezifische Funktionen 223, 230
  - Kurznamenoptionen 617
  - Nachrichten 244
  - Nucleotide-Schematabellen 240
  - PubMed-Schematabellen 235
  - Registrieren von Kurznamen 228
  - relationale Vergleichselemente 232
  - Serverdefinitionen registrieren 226
  - ungültige WHERE-Klauseln 232
  - unterstützte Versionen 6
  - Wrapper registrieren 225
  - Wrapperbibliotheksdateien 225
  - zu einem System zusammengeschlusener Datenbanken hinzufügen 222
  - Zugriff konfigurieren 222
- EnzymesByPathwyS, benutzerdefinierte Funktion
  - Syntax 553
- EnzymesByPathwyT, benutzerdefinierte Funktion
  - Syntax 554
- Erstellen von Wrappern
  - BioRS 81
  - BLAST 121
  - Dateien mit Tabellenstruktur 407
  - Datenquellen der DB2-Produktfamilie 182
  - Documentum 196
  - Entrez 225
  - Excel-Dateien 251
  - Extended Search 271
  - HMMER 299
  - Informix 319
  - ODBC 350
  - OLE DB 366
  - Oracle 377
  - Sybase 392
  - Teradata 421
  - XML 475
- Excel-Dateien
  - Beispielabfragen 254
  - Beispielbenutzerszenario 255
  - Beschreibung 249
  - CREATE NICKNAME, Syntax 609
  - CREATE SERVER, Anweisung 609
  - gültige Objekte für Kurznamen 18
  - Modell für die Dateizugriffsteuerung 256
  - Nachrichten 257
  - Serverdefinitionen registrieren 253
  - unterstützte Datentypen 581
  - unterstützte Versionen 6
  - Vorbereiten des Zugriffs 30
  - Wrapper registrieren 251
  - Wrapperbibliotheksdateien 252
  - zu einem System zusammengeschlusener Datenbanken hinzufügen 250
  - CREATE NICKNAME, Anweisung 253
  - Registrieren von Kurznamen 253
  - Zugriff konfigurieren 250
  - Zugriff über den ODBC-Wrapper 359

- Extended Search
  - Abfragerichtlinien 276
  - allgemeine Abfragesprache 281
  - Anpassen von Abfragen 275
  - Beispielabfragen 278
  - Benutzerzuordnungen registrieren 273
  - Beschreibung 263
  - CREATE FUNCTION, Anweisung 275, 616
  - CREATE NICKNAME, Anweisung 274
  - CREATE NICKNAME, Syntax 613
  - CREATE SERVER, Anweisung 611
  - CREATE USER MAPPING, Anweisung 273, 613
  - CREATE WRAPPER, Anweisung 611
  - durchsuchbare Quellen 263
  - Erstellen von mehrfachen Kurznamen 266
  - Funktionsschablonen registrieren 275
  - gültige Objekte für Kurznamen 18
  - Nachrichten 283
  - Registrieren von Kurznamen 274
  - Serverdefinitionen registrieren 272
  - unterstützte Datentypen 581
  - unterstützte Versionen 6
  - vertikale Tabellen 267
  - Wrapper registrieren 271
  - Wrapperbibliotheksdateien 272
  - zu einem System zusammengeschlusener Datenbanken hinzufügen 270
  - zugeordnete Felder 267
  - Zugriff konfigurieren 270

## F

- FaultQueue (Fehlerwarteschlange)
  - Konfiguration von Geschäftsobjektanwendungen 141
- FEDERATED, Parameter
  - einsetzen 50
- Fehlerbehebung
  - Informix-Serververbindungen testen 325
  - Microsoft SQL Server-Konfigurationen 345
  - ODBC-Konfigurationen 363
  - ODBC-Serververbindungen testen 356
  - Oracle-Konfigurationen 386
  - Oracle-Serververbindungen testen 382
  - Serververbindungen der DB2-Produktfamilie testen 188
  - Serververbindungen von Microsoft SQL Server testen 342
  - Sybase-Konfigurationen 402
  - Sybase-Serververbindungen testen 399
  - Teradata, Konfigurationen 430
  - Teradata-Serververbindungen testen 416, 426
  - Web-Services-Wrapper 468
- Fehlerprüfung
  - Web-Services-Wrapper 468
- Ferne Kataloginformationen 9



Fixierte Spalten  
 BLAST-Kurznamen 124  
 HMMER-Kurznamen 303

Flachdateien  
 siehe auch 'Dateien mit Tabellenstruktur' 6

FOLD\_ID, Serveroption  
 Definieren für Informix-Datenquellen 328  
 gültige Einstellungen 641  
 Werte, bei denen die Groß-/Kleinschreibung beachtet werden muss 26

FOLD\_PW, Serveroption  
 Definieren für Informix-Datenquellen 328  
 gültige Einstellungen 641  
 Werte, bei denen die Groß-/Kleinschreibung beachtet werden muss 26

Fremdschlüssel  
 mit Kurznamen verwenden 158

Funktionsschablonen  
 Extended Search 275

Funktionszuordnungen  
 Beschreibung 20  
 Optionen  
 gültige Einstellungen 707  
 planen 36

**G**

GenesByMotifsS, benutzerdefinierte Funktion  
 Syntax 577

GenesByPathwyS, benutzerdefinierte Funktion  
 Syntax 550

GenesByPathwyT, benutzerdefinierte Funktion  
 Syntax 551

GeneWise 526, 527

GeneWise, benutzerdefinierte Funktionen (UDFs)  
 Programmverbindung zur Bibliothek herstellen 525

Genomdatenbank 221

Geschäftsanwendungen  
 Adapter konfigurieren 141  
 Beispielabfragen 175  
 Serverdefinitionen 146  
 Sichten für zusammengeslossene Datenbanken 157  
 unterstützte Datentypen 581  
 Wrapper 145  
 Wrapperbibliotheksdateien 146  
 Zugriff über Wrapper 139

Geschäftsanwendungsdatenquellen  
 Registrieren von Kurznamen 148  
 Sichten für zusammengeslossene Datenbanken 450  
 zu einem System zusammengeslossener Datenbanken hinzufügen 144  
 Zugriff konfigurieren 144

Geschäftsobjektanwendungen  
 WebSphere Business Integration-Wrapper 137

Geschäftsobjektanwendungen, Wrapper  
 Beispiele 172, 459

Geschäftsobjekte 139  
 WebSphere Business Integration-Wrapper 137  
 zu einem System zusammengeslossener Datenbanken hinzufügen 144

Globaler Katalog  
 Beschreibung 9  
 Sichten mit zusammengeslossenen Informationen 629

## H

H2X\_PATH, HMMER-Dämon 291

High-Scoring Segment Pairs (HSP) 109

HMMER  
 CREATE SERVER, Anweisung (Beispiel) 300  
 fixierte Spalten für Kurznamen 303  
 Wrapperbibliotheksdateien 299

HMMER-Datenquelle  
 Abfragen 309  
 Beispielabfragen 308  
 Beschreibung 287  
 CREATE NICKNAME, Anweisung (Beispiel) 307

Dämon  
 konfigurieren 291  
 starten 296

Dämonkonfigurationsdatei, Beispiele 295  
 gültige Objekte für Kurznamen 18  
 hmmpfam, ausführbare Datei 291  
 Liste der Dienstprogramme 287  
 Nachrichten 311  
 Registrieren von Kurznamen 302  
 Serverdefinitionen registrieren 300  
 unterstützte Datentypen 581  
 unterstützte Versionen 6  
 Wrapper registrieren 299  
 zu einem System zusammengeslossener Datenbanken hinzufügen 290  
 Zugriff konfigurieren 290

HMMER\_OUT\_DIR\_PATH  
 HMMER-Dämon 291

HMMERPFAM\_PATH, HMMER-Dämon 291

hmmpfam, ausführbare Datei  
 Beschreibung 287  
 installierte Version prüfen 291

## I

IFILE, Serveroption  
 gültige Einstellungen 641

IGNORE\_UDT, Serveroption  
 gültige Einstellungen 641

Indekspezifikationen  
 Beschreibung 21

Informix  
 Benutzerzuordnungen registrieren 323  
 CREATE NICKNAME, Anweisung (Beispiel) 327  
 CREATE SERVER, Anweisung 322

Informix (*Forts.*)

CREATE USER MAPPING, Anweisung 324

Einrichten der Clientkonfigurationsdateien 314  
 gültige Objekte für Kurznamen 18  
 gültige Servertypen 635  
 Namen der Wrapperbibliotheksdatei 320  
 Optimierung 328  
 Registrieren von Kurznamen 326  
 Serverdefinitionen registrieren 321  
 Serververbindungen testen 325  
 standardmäßige progressive Typzuordnungen 679  
 standardmäßige regressive Typzuordnungen 695  
 Standardwrappername 14  
 Testen der Clientkonfigurationsdateien 314  
 Umgebungsvariablen einrichten 315  
 unterstützte Versionen 6  
 Wrapper registrieren 319  
 zu einem System zusammengeslossener Datenbanken hinzufügen 313  
 Zugriff konfigurieren 313

INFORMIX\_LOCK\_MODE, Serveroption  
 gültige Einstellungen 641

INITIAL\_INSTS, Funktionszuordnungsoption  
 gültige Einstellungen 707

INITIAL\_IOS, Funktionszuordnungsoption  
 gültige Einstellungen 707

INSTS\_PER\_ARGBYTE, Funktionszuordnungsoption  
 gültige Einstellungen 707

INSTS\_PER\_INVOC, Funktionszuordnungsoption  
 gültige Einstellungen 707

Integrierte Funktionen 20

IO\_RATIO, Serveroption  
 gültige Einstellungen 641

IOS\_PER\_ARGBYTE, Funktionszuordnungsoption  
 gültige Einstellungen 707

IOS\_PER\_INVOC, Funktionszuordnungsoption  
 gültige Einstellungen 707

IUD\_APP\_SVPT\_ENFORCE, Serveroption  
 gültige Einstellungen 641

## K

Katalog  
 siehe 'Globaler Katalog' 629

KEGG, benutzerdefinierte Funktionen  
 Beschreibungen von Argumenten 547  
 entfernen 577  
 Liste der Funktionen 546  
 registrieren 548  
 Übersicht 545  
 zurückgegebene SSDB-Spalten 560

Kennwörter  
 Werte, bei denen die Groß-/Kleinschreibung beachtet werden muss 26

Knoten  
 katalogisieren für Datenquellen der DB2-Produktfamilie 180

Kompensation, Beschreibung 11

Konfigurationen  
 zusammengeschlossene Datenquellen  
 Übersicht 76

Konfigurationen mit mehreren Partitionen 69

Kundenspezifische Funktionen  
 BioRS 80, 81, 88  
 Documentum 206  
 Entrez 223, 224, 230  
 Extended Search 275

Kurznamen  
 ändern  
 Einschränkungen 589  
 Kurznamenoptionen 592  
 lokale Spaltennamen 590  
 lokaler Datentyp 595  
 lokaler Datentyp, Beispiel 597  
 Spaltenoptionen 593  
 Übersicht 587

angeben  
 Spalten für nicht relationale Datenquellen 75

Beschreibung 17, 156, 449

Datenbanktechnologien 172, 459

Datenquellen der DB2-Produktfamilie 189

erstellen  
 BioRS 84  
 Dateien mit Tabellenstruktur 408  
 Entrez 228  
 Extended Search 266  
 für Kurznamen 75  
 Geschäftsanwendungsdatenquellen 148  
 HMMER 302  
 Informix 326  
 Microsoft SQL Server 344  
 ODBC 357  
 Oracle 384  
 Sybase 400  
 Teradata 427, 428  
 Web-Services 442, 451  
 WebSphere Business Integration-Datenquellen 148  
 XML 482

Geschäftsanwendungen 137  
 gültige Datenquellenobjekte 18  
 registrieren  
 Übersicht 73

Web Services Description Language (WSDL) 433

Kurznamen- und Spaltenoptionen  
 Web-Services-Wrapper 151, 433, 444  
 WebSphere Business Integration-Wrapper 151, 444  
 Kurznamenhierarchie, Beispiele 158

Kurznamenhierarchie, Beispiele 158

Kurznamenoptionen  
 angeben 592

Kurznamenoptionen  
 Beispiele 593  
 Beschreibung 19

Kurznamenstatistiken  
 BioRS 100, 101, 103

## L

LOGIN\_TIMEOUT, Serveroption  
 gültige Einstellungen 641

Lokaler Katalog  
 siehe 'Globaler Katalog' 9

LSBarCode, benutzerdefinierte Funktion  
 Beispiel 528  
 Syntax 528

LSDefineParse, benutzerdefinierte Funktion  
 Beispiele 520  
 Syntax 514

LSGeneWise, benutzerdefinierte Funktion 526, 527  
 Bibliothek 503

LSMultiMatch, benutzerdefinierte Funktion  
 Beispiel 530  
 Syntax 529

LSMultiMatch3, benutzerdefinierte Funktion  
 Beispiel 531  
 Syntax 531

LSNuc2Pep, benutzerdefinierte Funktion  
 Beispiel 538  
 Syntax 537

LSPatternMatch, benutzerdefinierte Funktion  
 Beispiel 521  
 Syntax 521

LSPep2AmbNuc, benutzerdefinierte Funktion  
 Beispiel 510  
 Fehlernachrichten 511  
 Syntax 508

LSPep2ProbNuc, benutzerdefinierte Funktion  
 Beispiel 512  
 Codon-Frequenztafel, Format 541  
 Fehlernachrichten 513  
 Syntax 511

LSPrositePattern, benutzerdefinierte Funktion  
 Beispiel 524  
 Syntax 523

LSRevComp, benutzerdefinierte Funktion  
 Beispiel 534  
 Syntax 533

LSRevNuc, benutzerdefinierte Funktion  
 Beispiel 535  
 Syntax 535

LSRevPep, benutzerdefinierte Funktion  
 Beispiel 536, 540  
 Syntax 536, 539

## M

MAX\_PENDING\_REQUESTS  
 BLAST-Dämon 115  
 HMMER-Dämon 291

Microsoft Excel  
 siehe Excel-Dateien 6

Microsoft SQL Server  
 Benutzerzuordnungen registrieren 341  
 CREATE NICKNAME, Anweisung (Beispiel) 345  
 CREATE SERVER, Anweisung 339  
 CREATE USER MAPPING, Anweisung (Beispiel) 341  
 Fehlerbehebung 345  
 gültige Objekte für Kurznamen 18  
 gültige Servertypen 635  
 Namen der Wrapperbibliotheksdatei 337  
 Optimierung 345  
 Registrieren von Kurznamen 344  
 Serverdefinitionen registrieren 338  
 Serververbindungen testen 342  
 standardmäßige progressive Typzuordnungen 679  
 standardmäßige regressive Typzuordnungen 695  
 Standardwrappernamen 14  
 Umgebungsvariablen einrichten 334  
 unterstützte Versionen 6  
 Vorbereiten von Servern mit zusammengeschlossenen Datenbanken 332  
 Wrapper registrieren 336  
 zu einem System zusammengeschlossener Datenbanken hinzufügen 331  
 Zugriff konfigurieren 331

MODULE, Wrapperoption  
 gültige Einstellungen 633

Motiv, benutzerdefinierte Funktionen (UDFs)  
 LSBarCode 528  
 LSMultiMatch 529  
 LSMultiMatch3 531

Mustererkennung, benutzerdefinierte Funktionen (UDFs)  
 LSPatternMatch 521  
 LSPrositePattern 523

## N

Nachrichten  
 BioRS-Wrapper 104  
 BLAST-Wrapper 134  
 djxlink 46  
 Documentum-Wrapper 214  
 Entrez-Wrapper 244  
 Excel-Wrapper 257  
 Extended Search-Wrapper 283  
 HMMER-Wrapper 311  
 LSPep2AmbNuc, benutzerdefinierte Funktion 511  
 LSPep2ProbNuc, benutzerdefinierte Funktion 513  
 Web-Services-Wrapper 468

Nachrichten (*Forts.*)  
 Wrapper für Dateien mit Tabellenstruktur 411  
 XML-Wrapper 491  
 Namenskonventionen  
 Objekte zusammengeschlüssener Datenbanken 25  
 Prüfliste für zusammengeschlüssene Datenquellen 36  
 NET8-Wrapper  
 Bibliotheksdateinamen, Oracle 378  
 Nicht relationale Datenquellen  
 Datentypzuordnungen angeben 20  
 Kurznamenspalten 75  
 unterstützte Datentypen 581  
 NODE, Serveroption (gültige Einstellungen) 641  
 Nucleotide  
 Datenbank 221  
 Schematabellen 240  
 NUMERIC\_STRING, Spaltenoption  
 Beispiel 593  
 gültige Einstellungen 671

## O

Objekte  
 benennen 25  
 ODA-Tool (Object Discovery Agent = Objekterkennungsagent)  
 Geschäftsanwendungen 139  
 Geschäftsanwendungswrapper 141  
 ODBC  
 Benutzerzuordnungen registrieren 354  
 CREATE NICKNAME, Anweisung (Beispiel) 358  
 CREATE SERVER, Anweisung 353  
 CREATE USER MAPPING, Anweisung 355  
 CREATE WRAPPER, Anweisung 351  
 Fehlerbehebung 363  
 gültige Objekte für Kurznamen 18  
 gültige Servertypen 635  
 Optimierung 363  
 Serverdefinitionen registrieren 352  
 Serververbindungen testen 356  
 standardmäßige progressive Typzuordnungen 679  
 Standardwrappernamen 14  
 unterstützte Versionen 6  
 Vorbereiten von Servern mit zusammengeschlüssenen Datenbanken 349  
 Wrapper registrieren 350  
 Wrapperbibliotheksdateien 351  
 zu einem System zusammengeschlüssener Datenbanken hinzufügen 347  
 Registrieren von Kurznamen 357  
 Zugriff konfigurieren 347  
 ODBC-Wrapper  
 Zugreifen auf Excel-Dateien 359  
 OLE DB  
 Benutzerzuordnungen registrieren 369  
 CREATE SERVER, Anweisung 368

OLE DB (*Forts.*)  
 CREATE USER MAPPING, Anweisung 370  
 gültige Servertypen 635  
 Serverdefinitionen registrieren 367  
 Standardwrappernamen 14  
 unterstützte Versionen 6  
 Wrapper registrieren 366  
 Wrapperbibliotheksdateien 367  
 zu einem System zusammengeschlüssener Datenbanken hinzufügen  
 Übersicht 365  
 Zugriff konfigurieren 365  
 OMIM-Datenbank 221  
 Optimierung  
 BLAST 134  
 Dateien mit Tabellenstruktur 411  
 Datenquellenkonfiguration  
 Informix 328  
 Microsoft SQL Server 345  
 ODBC 363  
 Oracle 386  
 Sybase 402  
 Teradata 430  
 Optimierungsprogramm  
 Beschreibung 10  
 Optionen  
 Kurznamen 661  
 Oracle  
 Beispiele für Anweisung CREATE USER MAPPING 381  
 Benutzerzuordnungen registrieren 381  
 Bibliotheksnamen 378  
 CREATE NICKNAME, Anweisung (Beispiel) 385  
 CREATE SERVER, Anweisung 379  
 Einrichten der Clientkonfigurationsdateien 376  
 Fehlerbehebung 386  
 gültige Objekte für Kurznamen 18  
 Optimierung 386  
 Serverdefinitionen registrieren 378  
 Serververbindungen testen 382  
 standardmäßige progressive Typzuordnungen 679  
 standardmäßige regressive Typzuordnungen 695  
 Standardwrappernamen 14  
 Testen der Clientkonfigurationsdateien 376  
 Umgebungsvariablen einrichten 372  
 Wrapper registrieren 377  
 Wrappernamen 378  
 zu einem System zusammengeschlüssener Datenbanken hinzufügen 371  
 Registrieren von Kurznamen 384  
 Zugriff konfigurieren 371

## P

PACKET\_SIZE, Serveroption  
 gültige Einstellungen 641  
 ParalogsByGeneS, benutzerdefinierte Funktion  
 Syntax 570

ParalogsByGeneT, benutzerdefinierte Funktion  
 Syntax 571  
 Parameter  
 FEDERATED 50  
 PASSWORD, Serveroption  
 gültige Einstellungen 641  
 Pathway-Datenbank  
 Liste der Funktionen 546  
 Pathway-Datenbank, benutzerdefinierte Funktionen (UDFs)  
 CompoundsByPathwyS 552  
 CompoundsByPathwyT 552  
 EnzymesByPathwyS 553  
 EnzymesByPathwyT 554  
 GenesByPathwyS 550  
 GenesByPathwyT 551  
 PathwysByCompndsS 557  
 PathwysByCompndsT, benutzerdefinierte Funktion 558  
 PathwysByEnzymesS 558  
 PathwysByEnzymesT 559  
 PathwysByGenesS 555  
 PathwysByGenesT 556  
 PathwysByCompndsS, benutzerdefinierte Funktion  
 Syntax 557  
 PathwysByCompndsT, benutzerdefinierte Funktion  
 Syntax 558  
 PathwysByEnzymesS, benutzerdefinierte Funktion  
 Syntax 558  
 PathwysByEnzymesT, benutzerdefinierte Funktion  
 Syntax 559  
 PathwysByGenesS, benutzerdefinierte Funktion  
 Syntax 555  
 PathwysByGenesT, benutzerdefinierte Funktion  
 Syntax 556  
 PCRE-Bibliothekspaket 524  
 PeopleSoft  
 Geschäftsobjekte 137  
 PERCENT\_ARGBYTES, Funktionszuordnungsoption  
 gültige Einstellungen 707  
 PLAN\_HINTS, Serveroption  
 gültige Einstellungen 641  
 Planen  
 Auswählen des korrekten Wrappers 29  
 Benutzerzuordnungen 34  
 Benutzerzuordnungen registrieren 72  
 Datentypzuordnungen 34  
 Funktionszuordnungen 36  
 nicht relational Kurznamenspalten 75  
 Prüfen der Konfiguration des Servers mit zusammengeschlüssenen Datenbanken 43  
 Prüfliste für die Konfiguration von Systemen zusammengeschlüssener Datenbanken 36  
 Registrieren von Kurznamen 73



Planen (*Forts.*)  
 Server mit zusammengeschlossenen Datenbanken  
   für Microsoft SQL Server 332  
   für ODBC 349  
 Serverdefinitionen registrieren 71  
 Umgebungsvariablen von Datenquellen 67  
 Wrapper registrieren 70  
 Zugreifen auf Excel-Daten 30

Primärschlüssel  
 mit Kurznamen verwenden 158

Programmverbindung herstellen (Linkedit)  
 siehe 'dxmlink' 43

Progressive Typzuordnungen  
 Standardzuordnungen 679

Prüflisten  
 Konfiguration von Systemen zusammengeschlossener Datenbanken planen 36

Pseudospalten  
 Documentum 200

PubMed-Datenbank 221

PubMed-Schematabellen 235

PUSHDOWN, Serveroption  
 gültige Einstellungen 641

Pushdown-Analyse  
 Beschreibung 10

## Q

Q\_SEQ\_DIR\_PATH  
 BLAST-Dämon 115  
 HMMER-Dämon 291

## R

Reguläre Ausdrücke, Unterstützung  
 Benutzerdefinierte Life Sciences-Funktionen 524

REMOTE\_AUTHID, Benutzeroption  
 gültige Einstellungen 659

REMOTE\_DOMAIN, Benutzeroption  
 gültige Einstellungen 659

REMOTE\_NAME, Funktionszuordnungsoption  
 gültige Einstellungen 707

REMOTE\_PASSWORD, Benutzeroption  
 gültige Einstellungen 659

RequestQueue (Anforderungswarteschlange)  
 Konfiguration von Geschäftsobjektanwendungen 141

ResponseQueue (Antwortwarteschlange)  
 Konfiguration von Geschäftsobjektanwendungen 141

RevBestNbrsByGeneS, benutzerdefinierte Funktion  
 Syntax 568

RevBestNbrsByGeneT, benutzerdefinierte Funktion  
 Syntax 569

Rückübersetzung, benutzerdefinierte Funktionen (UDFs)  
 LSPep2AmbNuc 508

Rückübersetzung, benutzerdefinierte Funktionen (UDFs) (*Forts.*)  
 LSPep2ProbNuc 511  
 Übersicht 508

RUNPFAM\_PATH  
 HMMER-Dämon 291

## S

SAP  
 Geschäftsobjekte 137

Schablonen  
 Geschäftsobjekte 139

ScoreBetweenGenesS, benutzerdefinierte Funktion  
 Syntax 575

Sequence Similarity-Datenbank  
 Liste der Funktionen 546  
 zurückgegebene Spalten 560

Sequence Similarity-Datenbank, benutzerdefinierte Funktionen (UDFs)  
 AllNbrsByGeneS 561  
 AllNbrsByGeneT 562  
 BestHmlgsByGenesS 572  
 BestHmlgsByGenesT 573  
 BestNbrsByGeneS 566  
 BestNbrsByGeneT 567  
 BstBstHmlgByGenesS 573  
 BstBstHmlgByGenesT 574  
 BstBstNbrsByGeneS 564  
 BstBstNbrsByGeneT 565  
 DefinitionsByGeneS 576  
 GenesByMotifsS 577  
 ParalogsByGeneS 570  
 ParalogsByGeneT 571  
 RevBestNbrsByGeneS 568  
 RevBestNbrsByGeneT 569  
 ScoreBetweenGenesS 575

Server mit zusammengeschlossenen Datenbanken 5  
 Beschreibung 4  
 mehrere Server konfigurieren 66  
 Prüfen der Konfiguration 43  
 Wrapper 13  
 Wrappermodule 13

Serverdefinitionen  
 Beschreibung 15  
 BioRS 83  
 BLAST 122  
 Dateien mit Tabellenstruktur 408  
 Datenquellen der DB2-Produktfamilie 184  
 Documentum 198  
 Entrez 226  
 Excel-Dateien 253  
 Extended Search 272  
 Geschäftsanwendungen 146  
 HMMER 300  
 Informix 321  
 Microsoft SQL Server 338  
 ODBC 352  
 OLE DB 367  
 Oracle 378  
 Sybase 394  
 Teradata 422  
 Übersicht zur Registrierung 71  
 Web-Services 441

Serverdefinitionen (*Forts.*)  
 WebSphere Business Integration-Wrapper 146  
 XML 477

Serveroptionen  
 Beschreibung 15  
 gültige Einstellungen 641  
 temporäre 15

Servertypen  
 gültige Typen für zusammengeschlossene Datenbanken 635

SET SERVER OPTION, Anweisung  
 Option temporär festlegen 15

Sicherheit  
 Web-Services-Wrapper 467

Sichten  
 für zusammengeschlossene Datenbanken für Nicht-Rootkurznamen 487  
 Geschäftsanwendungen 157, 450

Sichten für zusammengeschlossene Datenbanken  
 für Nicht-Rootkurznamen 487  
 Geschäftsanwendungen 450

Siebel  
 Geschäftsobjekte 137

Sortierfolgen  
 Systeme zusammengeschlossener Datenbanken 51, 52, 55

Spaltennamen  
 ändern 590

Spaltenoptionen  
 Beschreibung 19  
 für Kurznamen angeben 593  
 gültige Einstellungen 671

SQL-Compiler  
 in einem System zusammengeschlossener Datenbanken 10

SQL-Programmversion  
 Beschreibung 11

SQLNET-Wrapper  
 Bibliotheksdateinamen 378

Statistikdaten  
 Datenquelle aktualisieren 28  
 für BioRS 99, 100  
 Prüfliste für zusammengeschlossene Datenquellen 36

Steuerzentrale  
 Datenquellen konfigurieren 63, 64  
 Schnittstelle für Systeme zusammengeschlossener Datenbanken 22

Sybase  
 Beispiele für Anweisung CREATE USER MAPPING 398  
 Benutzerzuordnungen registrieren 397  
 CREATE NICKNAME, Anweisung (Beispiel) 401  
 CREATE SERVER, Anweisung 395  
 Einrichten der Clientkonfigurationsdateien 391  
 Fehlerbehebung 402  
 gültige Objekte für Kurznamen 18  
 gültige Servertypen 635  
 Namen der Wrapperbibliotheksdatei 393  
 Optimierung 402  
 Serverdefinitionen registrieren 394

Sybase (*Forts.*)  
 Serververbindungen testen 399  
 standardmäßige progressive Typzuordnungen 679  
 standardmäßige regressive Typzuordnungen 695  
 Standardwrappernamen 14  
 Testen der Clientkonfigurationsdateien 391  
 Umgebungsvariablen einrichten 388  
 unterstützte Versionen 6  
 Wrapper registrieren 392  
 zu einem System zusammengeschlossener Datenbanken hinzufügen 387  
 Registrieren von Kurznamen 400  
 Zugriff konfigurieren 387

Syntax  
 benutzerdefinierte KEGG-Funktionen 547

SYSCAT, Katalogsichten 629

SYSSTAT, Katalogsichten 629

Systeme zusammengeschlossener Datenbanken  
 Übersicht 3

Szenarien  
 Zugreifen auf Excel-Dateien 255

## T

Teradata  
 Benutzerzuordnungen registrieren 425  
 Binden zur Laufzeit aktivieren (AIX) 417, 430  
 CREATE NICKNAME, Anweisung (Beispiel) 429  
 CREATE SERVER, Anweisung 424  
 CREATE USER MAPPING, Anweisung 425  
 Fehlerbehebung 430  
 gültige Objekte für Kurznamen 18  
 gültige Servertypen 635  
 Kurznamen für Server mit zusammengeschlossenen Datenbanken, Übersicht 427  
 Namen der Wrapperbibliotheksdatei 422  
 Optimierung 430  
 Serverdefinitionen registrieren 422  
 Serververbindungen testen 416, 426  
 standardmäßige progressive Typzuordnungen 679  
 standardmäßige regressive Typzuordnungen 695  
 Standardwrappernamen 14  
 Umgebungsvariablen einrichten 418  
 Wrapper registrieren 421  
 zu einem System zusammengeschlossener Datenbanken hinzufügen 415  
 Binden zur Laufzeit prüfen (AIX) 417  
 Registrieren von Kurznamen 428  
 Zugriff konfigurieren 415  
 Zugriffsprotokollierung 430

TIMEFORMAT, Serveroption  
 gültige Einstellungen 641

TIMEOUT, Serveroption  
 gültige Einstellungen 641

TIMESTAMPFORMAT, Serveroption  
 gültige Einstellungen 641

TurboBlast 130

## U

Übersetzungsfunktionen, benutzerdefiniert  
 LSNuc2Pep 537  
 LSTransAllFrames 539

Übersetzungstabelle  
 Benutzerdefinierte Life Sciences-Funktionen  
 Beispiel 543  
 Format 542

Umgebungsvariablen  
 Documentum 195  
 in einem Exemplar zusammengeschlossener Datenbanken mit mehreren Partitionen 69  
 Informix 315  
 Microsoft SQL Server 334  
 Oracle 372  
 Sybase 388  
 Teradata 418  
 Übersicht zu Systemen zusammengeschlossener Datenbanken 67

Umkehrfunktionen, benutzerdefiniert  
 LSRevComp 533  
 LSRevNuc 535  
 LSRevPep 536

Unicode 56

Unterstützung in der Landessprache (NLS) 51

## V

VARCHAR\_NO\_TRAILING\_BLANKS, Serveroption  
 gültige Einstellungen 641

VARCHAR\_NO\_TRAILING\_BLANKS, Spaltenoption  
 Beispiel 593  
 gültige Einstellungen 671

Vergleichselemente  
 Entrez 232

Verteiltes Datenbankverwaltungssystem 3

Vertikale Tabellen  
 Extended Search 267

## W

Web-Services  
 Beispielabfragen 463  
 Beispiele 172, 459  
 Fehlerbehebung 468  
 Kurznamen 433, 451  
 Registrieren von Kurznamen 442  
 Serverdefinitionen registrieren 441  
 unterstützte Datentypen 581  
 Wrapperbibliotheksdateien 441  
 zu einem System zusammengeschlossener Datenbanken hinzufügen 439

Web-Services (*Forts.*)  
 Zugriff konfigurieren 439

Web Services Description Language (WSDL)  
 Kurznamen 433  
 Zuordnen zu Kurznamen 156, 449

Web-Services-Provider  
 Fehlerbehebung 468

Web-Services-Wrapper  
 erstellen 440

WebSphere Business Integration  
 Konfiguration 139

WebSphere Business Integration-Adapter  
 installieren 141

WebSphere Business Integration-Wrapper 137  
 Beispiele 158, 172, 459  
 erstellen 145  
 Registrieren von Kurznamen 148  
 Serverdefinitionen 146  
 unterstützte Datentypen 581  
 Wrapperbibliotheksdateien 146

WebSphere MQ  
 Nachrichtenwarteschlangen 141

Wiederholung von Spaltenattributen  
 Documentum 205

Wrapper  
 Beschreibung 13  
 Bibliotheken 44  
 Bibliotheksdateien 45  
 Geschäftsanwendungen 137, 145  
 planen 29  
 Prüfliste für zusammengeschlossene Datenquellen 36  
 registrieren für BioRS 81  
 registrieren für BLAST 121  
 registrieren für Dateien mit Tabellenstruktur 407  
 registrieren für Datenquellen der DB2-Produktfamilie 182  
 registrieren für Documentum 196  
 registrieren für Entrez 225  
 registrieren für Excel-Dateien 251  
 registrieren für Extended Search 271  
 registrieren für HMMER 299  
 registrieren für Informix 319  
 registrieren für Microsoft SQL Server 336  
 registrieren für ODBC 350  
 registrieren für OLE DB 366  
 registrieren für Oracle 377  
 registrieren für Sybase 392  
 registrieren für Teradata 421  
 registrieren für XML 475  
 Standardnamen 14  
 Übersicht zur Registrierung 70  
 Web-Services-Datenquellen 439, 440  
 WebSphere Business Integration 145

Wrapperbibliotheken 441  
 BioRS-Dateinamen 82  
 BLAST-Dateinamen 122  
 Dateinamen der DB2-Produktfamilie 183  
 Dateinamen von WebSphere Business Integration-Wrappern 146  
 Documentum-Dateinamen 197  
 Entrez-Dateinamen 225

- Wrapperbibliotheken (*Forts.*)
  - Excel-Dateinamen 252
  - Extended Search-Dateinamen 272
  - HMMER-Dateinamen 299
  - Informix-Dateinamen 320
  - manuelles Herstellen der Programm-  
verbindung 47
  - Microsoft SQL Server-Dateina-  
men 337
  - Namen von Dateien mit Tabellen-  
struktur 407
  - ODBC-Dateinamen 351
  - OLE DB-Dateinamen 367
  - Oracle-Dateinamen 378
  - Sybase-Dateinamen 393
  - Teradata-Dateinamen 422
  - XML-Dateinamen 476
- Wrapperoptionen
  - gültige Einstellungen 633
- Zusammengeschlossene Datenbank
  - Sortierfolge 52, 55
- Zusammengeschlossene Datenbanken
  - Beschreibung 8
  - erstellen 58
  - Systemkatalog 9

## X

- XML
  - Aufwandsmodellfunktion 481, 488
  - Beispielabfragen 489
  - Beschreibung 471
  - CREATE NICKNAME, Anweisung  
(Beispiel) 482
  - CREATE NICKNAME, Syntax 623
  - Erstellen von Sichten für zusammen-  
geschlossene Datenbanken für  
Nicht-Stammkurznamen 487
  - gültige Objekte für Kurznamen 18
  - Nachrichten 491
  - Quellendokumente und Kurz-  
namen 479
  - Serverdefinitionen registrieren 477
  - unterstützte Datentypen 581
  - unterstützte Versionen 6
  - Wrapper registrieren 475
  - Wrapperbibliotheksdateien 476
  - zu einem System zusammengeschlos-  
sener Datenbanken hinzufügen 474
    - CREATE NICKNAME, Anwei-  
sung 482
    - Registrieren von Kurznamen 479,  
482
  - Zugriff konfigurieren 474
- XML-Schemata
  - Geschäftsanwendungen 139
  - Zuordnen zu Kurznamen 156, 449
- XPath
  - mit Kurznamen verwenden 156, 449

## Z

- Zerleinerte Geschäftsobjekte
  - Kurznamen 137
- Zugeordnete Felder
  - Extended Search 267
- Zugriffsmöglichkeiten
  - Funktionen 719
- Zugriffspläne
  - Beschreibung 10
- Zuordnen
  - XML-Dokumente 156, 449

---

## Kontaktaufnahme mit IBM

Telefonische Unterstützung erhalten Sie über folgende Nummern:

- Unter 0180 3 313233 erreichen Sie Hallo IBM, wo Sie Antworten zu allgemeinen Fragen erhalten.
- Unter 0190 7 72243 erreichen Sie die DB2 Helpline, wo Sie Antworten zu DB2-spezifischen Problemen erhalten.

Informationen zur nächsten IBM Niederlassung in Ihrem Land oder Ihrer Region finden Sie im IBM Verzeichnis für weltweite Kontakte, das Sie im Web unter [www.ibm.com/planetwide](http://www.ibm.com/planetwide) abrufen können.

---

## Produktinformationen

Informationen zu DB2 Information Integrator erhalten Sie telefonisch oder im World Wide Web.

Telefonische Unterstützung erhalten Sie über folgende Nummern:

- Unter 0180 3 313233 erreichen Sie Hallo IBM, wo Sie Antworten zu allgemeinen Fragen erhalten.
- Unter 0180 5 5090 können Sie Handbücher telefonisch bestellen.

Rufen Sie im Web die Site

[www.ibm.com/software/data/integration/db2ii/support.html](http://www.ibm.com/software/data/integration/db2ii/support.html) auf. Auf dieser Site finden Sie Informationen zu folgenden Themen:

- Bibliothek mit technischer Dokumentation
- Bestellen von Büchern
- Client-Downloads
- Newsgroups
- FixPaks
- Neuerungen
- Links auf Webressourcen

---

## Kommentare zur Dokumentation

Ihr Feedback unterstützt IBM bei der Bereitstellung qualitativ hochwertiger Informationsmaterialien. Bitte senden Sie uns Ihre Kommentare zum vorliegenden Handbuch oder zu anderen DB2 Information Integrator-Dokumentationen. Zur Abgabe von Kommentaren können Sie folgendermaßen vorgehen:

- Verwenden Sie für Ihren Kommentar das Onlineformular für Leserkommentare, das unter [www.ibm.com/software/data/rcf](http://www.ibm.com/software/data/rcf) bereitgestellt wird.
- Senden Sie Ihre Kommentare per E-Mail an [comments@us.ibm.com](mailto:comments@us.ibm.com). Bitte geben Sie den Namen des Produkts, seine Versionsnummer sowie den Titel und die Teilenummer (sofern vorhanden) der Veröffentlichung an, auf die sich Ihr Kommentar bezieht. Geben Sie bei Kommentaren zu einer spezifischen Textstelle bitte auch die Position dieser Textstelle (z. B. Abschnittsüberschrift, Abbildungs- oder Seitennummer) innerhalb der Veröffentlichung an.





**IBM**