

IBM DB2 Information Integrator



Systeme zusammengeschlossener Datenbanken

Version 8.2

IBM DB2 Information Integrator



Systeme zusammengeschlossener Datenbanken

Version 8.2

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die allgemeinen Informationen unter „Bemerkungen“ auf Seite 359 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 Federated Systems Guide Version 8.2,

IBM Form SC18-7364-01,

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

Juli 2004

Inhaltsverzeichnis

Zu diesem Handbuch	ix
Zielgruppe	ix

Teil 1. Einführung 1

Kapitel 1. Überblick über ein System zusammengeschlossener Datenbanken . 3

System zusammenschlossener Datenbanken	3
Der Server mit zusammenschlossenen Datenbanken	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	15
Serverdefinitionen und Serveroptionen	16
Benutzerzuordnungen	17
Kurznamen und Datenquellenobjekte	17
Gültige Datenquellenobjekte	19
Kurznamenspaltoptionen	20
Datentypzuordnungen	21
Funktionszuordnungen	21
Indespezifikationen	22
Sortierfolgen	23
Bestimmen der Sortierreihenfolge durch die Sortierfolge	23
Definieren der lokalen Sortierfolge zum Optimieren von Abfragen	24
Interaktion mit einem System zusammenschlossener Datenbanken	25
DB2-Befehlszeilenprozessor (CLP)	25
DB2-Befehlszentrale	26
DB2-Steuerzentrale	26
Anwendungsprogramme	27
DB2-Produktfamilie	27
Provider für Web-Services	27

Teil 2. Verwaltung und Pflege 29

Kapitel 2. Ändern von Datenquellenkonfigurationen 31

Ändern eines Wrappers	31
Ändern eines Wrappers - Beispiele	32
Ändern von Serverdefinitionen und Serveroptionen	32
Ändern von Serverdefinitionen und Serveroptionen - ausführliche Informationen	34
Ändern der Datenquellenversion in einer Serverdefinition	34

Ändern aller Serverdefinitionen für einen bestimmten Datenquellentyp	35
Verwenden von Serveroptionen in Serverdefinitionen	35
Ändern einer Benutzerzuordnung	37
Ändern eines Kurznamens	39
Ändern von Kurznamen - ausführliche Informationen	41
Einschränkungen zur Änderung von Kurznamen	41
Ändern der Spaltennamen von Kurznamen	43
Ändern der Optionen von Kurznamen	44
Ändern der Spaltenoptionen von Kurznamen	45
Löschen eines Wrappers	47
Löschen einer Serverdefinition	49
Löschen einer Benutzerzuordnung	50
Löschen eines Kurznamens	51

Kapitel 3. Datentypzuordnungen 53

Datentypzuordnungen in einem System zusammenschlossener Datenbanken	53
Datentypzuordnungen und der globale Katalog zusammenschlossener Datenbanken	54
Zeitpunkt für die Erstellung alternativer Datentypzuordnungen	55
Datentypzuordnungen für nicht relationale Datenquellen	56
Progressive und regressive Datentypzuordnungen	56
Erstellen von Datentypzuordnungen	57
Erstellen einer Datentypzuordnung für einen bestimmten Datentyp einer Datenquelle - Beispiel	58
Erstellen einer Typzuordnung für den Datentyp und die Version einer Datenquelle - Beispiel	59
Erstellen einer Typzuordnung für alle Datenquellenobjekte auf einem bestimmten Server - Beispiel	60
Ändern eines lokalen Typs für ein Datenquellenobjekt	61
Ändern eines lokalen Typs für ein Datenquellenobjekt - Beispiele	63
Ändern langer Datentypen in VARCHAR-Datentypen	64

Kapitel 4. Funktionszuordnungen und benutzerdefinierte Funktionen 67

Funktionszuordnungen in einem System zusammenschlossener Datenbanken	67
Erstellen eigener Funktionszuordnungen	67
Bedeutung von Funktionszuordnungen	68
Funktionsweise von Funktionszuordnungen in einem System zusammenschlossener Datenbanken	69
Voraussetzungen für die Zuordnung von benutzerdefinierten Funktionen (UDFs, User-Defined Functions)	70
Funktionsschablonen	70
Erstellen von Funktionsschablonen	71

Bereitstellen von Informationen zum Funktionszuordnungsaufwand für das Abfrageoptimierungsprogramm	73
Aktualisieren der Aufwandsinformationen	75
Angaben von Funktionsnamen in einer Funktionszuordnung	75
Erstellen von Funktionszuordnungen	76
Erstellen von Funktionszuordnungen - ausführliche Informationen	77
Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp	77
Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp und eine bestimmte Datenquellentypversion	78
Erstellen einer Funktionszuordnung für alle Datenquellenobjekte auf einem bestimmten Server	79
Benutzerdefinierte Funktionen in Anwendungen	80
Inaktivieren einer Standardfunktionszuordnung	81
Löschen einer benutzerdefinierten Funktionszuordnung	82

Kapitel 5. Indexspezifikationen 83

Indexspezifikationen in einem System zusammengesetzter Datenbanken	83
Erstellen von Indexspezifikationen für Datenquellenobjekte	84
Erstellen von Indexspezifikationen für Tabellen, die neue Indizes erhalten	86
Erstellen von Indexspezifikationen für Sichten	88
Erstellen von Indexspezifikationen für Informix-Synonyme	89

Kapitel 6. Transparente DDL 93

Was ist transparente DDL?	93
Transparente DDL - Einschränkungen	94
Ferne LOB-Spalten und transparente DDL	95
Erstellen neuer ferner Tabellen unter Verwendung von transparenter DDL	95
Ändern von fernen Tabellen mit transparenter DDL	99
Löschen von fernen Tabellen mit transparenter DDL	101

Kapitel 7. Transaktionsunterstützung in einem System zusammengesetzter Datenbanken 103

Transaktionsunterstützung in einem System zusammengesetzter Datenbanken	103
Was ist eine Aktualisierung in einem System zusammengesetzter Datenbanken?	105
Was ist eine Aktualisierungstransaktion in einer Durchgriffssitzung?	106
Transaktionsunterstützung bei transparenter DDL	106
Datenquellen, die DDL-Anweisungen automatisch festschreiben	107
Benutzerdefinierte Funktionen, die im Push-down-Modus zur Verarbeitung an die Datenquelle weitergeleitet werden	107

Kapitel 8. INSERT-, UPDATE- und DELETE-Operationen 109

Zugriffsrechte für die Anweisungen INSERT, UPDATE und DELETE	109
Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammengesetzter Datenbanken	110
Nicht unterstützte Datenquellen	110
Referenzielle Integrität in einem System zusammengesetzter Datenbanken	111
Anweisungen INSERT, UPDATE und DELETE für große Objekte (LOBs)	112
Gewährleisten der Anweisungsautarkie in einem System zusammengesetzter Datenbanken	112
Arbeiten mit Kurznamen	114
WITH HOLD-Syntax	115
Auslöser	115
Funktionsweise von Kurznamen - ausführliche Informationen	116
Mit Kurznamen verwendbare SQL-Anweisungen	116
Zugreifen auf neue Datenquellenobjekte	121
Zugreifen auf Datenquellen mit Durchgriffssitzungen	123
Zugreifen auf heterogene Daten durch Sichten für zusammengesetzte Datenbanken	124
Erstellen eines Kurznamens für einen Kurznamen	126
Auswählen von Daten in einem System zusammengesetzter Datenbanken	126
Ändern von Daten in einem System zusammengesetzter Datenbanken	130
Einfügen von Daten in Datenquellenobjekte	130
Aktualisieren von Daten in Datenquellenobjekten	131
Löschen von Daten aus Datenquellenobjekten	132
Zuordnungssemantik in einem System zusammengesetzter Datenbanken	133
Zuordnungssemantik in einem System zusammengesetzter Datenbanken - Beispiele	136

Kapitel 9. Überwachen eines Systems zusammengesetzter Datenbanken 137

Diagnoseanzeiger für Kurznamen und Server zusammengesetzter Datenbanken	137
Aktivieren der Diagnoseanzeiger von Systemen zusammengesetzter Datenbanken	139
Überwachen des Status von Kurznamen und Servern für zusammengesetzte Datenbanken	139
Überwachen des Status von Kurznamen und Servern für zusammengesetzte Datenbanken - Beispiel	140
Ausführen des Snapshot Monitor für Systeme zusammengesetzter Datenbanken - Übersicht	141
Überwachen von Abfragefragmenten für zusammengesetzte Datenbanken	142
Ausführen des Snapshot Monitor für Abfragefragmente für zusammengesetzte Datenbanken - Beispiel	142

Kapitel 10. Unicode-Unterstützung für Datenquellen zusammengesetzter Datenbanken 145

Unicode-Unterstützung für Systeme zusammengesetzter Datenbanken 145
Angaben der Client-Codepage für die Unicode-Unterstützung von Microsoft SQL Server- und ODBC-Datenquellen 147
Unterstützte Unicode-Codepages für die Option CODEPAGE des MSSQL- und ODBC-Wrappers . . . 148
Angaben der Dateicodepage für die Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur 149
Angaben der Dateicodepage für Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur - Beispiel 149
Fehler auf Grund des Unterschieds zwischen der Größe des fernen Codepunkts und der Größe des Codepunkts für zusammengesetzte Datenbanken 150

Teil 3. Leistung 151

Kapitel 11. Optimieren der Leistung eines Systems zusammengesetzter Datenbanken 153

Veröffentlichungen zur Leistung zusammengesetzter Datenbanken 153
Optimieren der Abfrageverarbeitung 153
Pushdown-Analyse 156
Pushdown-Analyse - ausführliche Informationen
 Serverkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten 158
 Kurznamenkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten 163
 Abfragekenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten 165
Entscheidungen bei der Pushdown-Analyse . . . 165
 Analysieren des Auswertungsortes einer Abfrage 165
 Entscheidungen für die Auswertung von Zugriffsplänen 167
 Aktualisieren und Anpassen von Datenquellen 169
Globale Optimierung 170
Globale Optimierung - ausführliche Informationen 171
 Serverkenndaten mit Auswirkungen auf die globale Optimierung 171
 Kurznamenkenndaten mit Auswirkungen auf die globale Optimierung 173
Entscheidungen der globalen Optimierung . . . 176
 Analysieren der globalen Optimierung . . . 176
 Durch den Zugriffsplan bestimmte Möglichkeiten der Optimierung 178
Systemmonitorelemente, die die Leistung beeinflussen 179

Kapitel 12. Parallelität bei Abfragen, die auf Kurznamen verweisen 181

Parallelität bei Abfragen, die auf Kurznamen verweisen 181
Partitionsinterne Parallelität bei Abfragen, die auf Kurznamen verweisen 182
Aktivieren der partitionsinternen Parallelität bei Abfragen, die auf Kurznamen verweisen 183
Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen 183
Aktivieren von partitionsübergreifender Parallelität bei Abfragen, die auf Kurznamen verweisen . . . 186
Rechenpartitionsgruppen 186
Definieren einer Rechenpartitionsgruppe 187
Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen - erwartetes Leistungsverhalten 188
Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen 188
Aktivieren von gemischter Parallelität bei Abfragen, die auf Kurznamen verweisen 189
Parallele Zugriffspläne von Abfragen, die auf Kurznamen verweisen 189
Partitionsinterne Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne. 190
Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne 192
Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne . . 196

Kapitel 13. Gespeicherte Abfragetabellen und Systeme zusammengesetzter Datenbanken 199

Gespeicherte Abfragetabellen und Systeme zusammengesetzter Datenbanken - Übersicht . . . 199
Erstellen einer gespeicherten Abfragetabelle in Systemen zusammengesetzter Datenbanken . . . 200
Datenquellenspezifische Einschränkungen für gespeicherte Abfragetabellen 201
Einschränkungen bei der Verwendung gespeicherter Abfragetabellen mit Kurznamen 204

Kapitel 14. Cachetabellen in einem System zusammengesetzter Datenbanken 205

Cachetabellen 205
Erstellen einer Cachetabelle. 207
Aktivieren eines Caches 208
Hinzufügen einer gespeicherten Abfragetabelle zu einer Cachetabelle 209
Löschen einer gespeicherten Abfragetabelle aus einer Cachetabelle 210
Löschen einer Cachetabelle 210

Kapitel 15. Informative Integritätsbedingungen für Kurznamen in einem System zusammengesetzter Datenbanken 213

Informative Integritätsbedingungen für Kurznamen 213

Angeben von informativen Integritätsbedingungen für Kurznamen	213
Angeben von informativen Integritätsbedingungen für Kurznamen - Beispiele	215
 Kapitel 16. Kurznamenstatistiken	219
Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht	219
Abrufen von Kurznamenstatistiken	220
Abrufen von Kurznamenstatistiken von der Befehlszeile aus - Beispiele	222
Erstellen eines DB2-Toolskatalogs.	223
Anzeigen des Aktualisierungsstatus der Kurznamenstatistiken	224

Teil 4. Anwendungsprogrammierung 225

 Kapitel 17. Anwendungsprogrammierungsszenario	227
--	------------

Kapitel 18. Anwendungsprogrammierung für Systeme zusammengeschlossener Datenbanken 229

Interaktion von Clientanwendungen mit Datenquellen	229
Arbeiten mit Kurznamen in Ihren Anwendungen	230
Verweisen auf Datenquellenobjekte in SQL-Anweisungen über Kurznamen	230
Kurznamen in DDL-Anweisungen	231
Datenquellenstatistiken - Anwendungsauswirkungen	232
Kurznamen zum Aufrufen gespeicherter Prozeduren	233
Definieren von Spaltenoptionen für Kurznamen	233
Erstellen und Verwenden von Sichten für zusammengeschlossene Datenbanken	234
Verwenden von Isolationsstufen zur Gewährleistung der Datenintegrität.	235
LOB-Unterstützung in einem System zusammengeschlossener Datenbanken	237
LOB-Unterstützung in einem System zusammengeschlossener Datenbanken - ausführliche Informationen.	239
LOB-Querverweise	239
Einschränkungen für LOBs	239
Verteilte Anforderungen zur Abfrage von Datenquellen	239
Optimieren verteilter Anforderungen mit Serveroptionen	241
Verwenden von Durchgriffssitzungen in Anwendungen	242
Direktes Abfragen von Datenquellen im Durchgriffsmodus	242
Überlegungen und Einschränkungen bei der Verwendung des Durchgriffsmodus in Systemen zusammengeschlossener Datenbanken	243
Durchgriffssitzungen zu Oracle-Datenquellen	244

Teil 5. Referenz 245

 Kapitel 19. Sichten in der globalen Katalogtabelle mit Informationen der zusammengeschlossenen Datenbank	247
---	------------

 Kapitel 20. Wrapperoptionen für Systeme zusammengeschlossener Datenbanken	251
--	------------

 Kapitel 21. Serveroptionen für Systeme zusammengeschlossener Datenbanken	253
---	------------

 Kapitel 22. Benutzerzuordnungsoptionen für Systeme zusammengeschlossener Datenbanken	271
---	------------

 Kapitel 23. Kurznamenoptionen für Systeme zusammengeschlossener Datenbanken	273
--	------------

 Kapitel 24. Kurznamenspaltenoptionen für Systeme zusammengeschlossener Datenbanken	283
---	------------

 Kapitel 25. Funktionszuordnungsoptionen für Systeme zusammengeschlossener Datenbanken	291
--	------------

Kapitel 26. Gültige Servertypen in SQL-Anweisungen 293

BioRS-Wrapper	293
BLAST-Wrapper	293
CTLIB-Wrapper	294
Documentum-Wrapper	294
DRDA-Wrapper	294
Entrez-Wrapper	295
Excel-Wrapper	295
Extended Search-Wrapper	295
HMMER-Wrapper	295
Informix-Wrapper	296
MSSQLODBC3-Wrapper.	296
NET8-Wrapper	296
ODBC-Wrapper	296
OLE DB-Wrapper	296
Wrapper für Dateien mit Tabellenstruktur	296
Teradata-Wrapper	297
Wrapper für Web-Services	297
WebSphere Business Integration-Wrapper	297
XML-Wrapper	297

 Kapitel 27. Standardmäßige progressive Datentypzuordnungen	299
---	------------

Datenquellen von DB2 für z/OS und OS/390	299
--	-----

Datenquellen von DB2 für iSeries.	300	Vom XML-Wrapper unterstützte Datentypen	337
Datenquellen von DB2 Server für VM und VSE	302		
Datenquellen von DB2 für Linux, UNIX und Windows	303	Kapitel 31. Systemmonitorelemente der zusammengesetzten Datenbanken	339
Informix-Datenquellen	304		
Microsoft SQL Server-Datenquellen	305	Kapitel 32. Gespeicherte Prozedur SYSPROC.NNSTAT	341
ODBC-Datenquellen	308		
Oracle NET8-Datenquellen	309	Kapitel 33. HADR und Datenquellen in zusammengesetzten Datenbanken.	343
Sybase-Datenquellen	310		
Teradata-Datenquellen	311	Kapitel 34. Gateway-Server-Informationen für Steuerkomponententraps abfragen	345
Kapitel 28. Standardmäßige regressive Typzuordnungen	315	DB2 Information Integrator-Dokumentation	347
Datenquellen von DB2 für z/OS und OS/390	316	Zugriff auf die DB2 Information Integrator-Dokumentation	347
Datenquellen von DB2 für iSeries.	317	Dokumentation zur Replikationsfunktion unter z/OS	349
Datenquellen von DB2 für VM und VSE	318	Dokumentation zur Event-Publishing-Funktion für DB2 Universal Database unter z/OS.	350
Datenquellen von DB2 für Linux, UNIX und Windows	319	Dokumentation zur Event-Publishing-Funktion für IMS und VSAM unter z/OS	351
Informix-Datenquellen	320	Dokumentation zur Event-Publishing-Funktion und Replikationsfunktion unter Linux, UNIX und Windows	352
Microsoft SQL Server-Datenquellen	321	Dokumentation zur Funktion für Systeme zusammengesetzter Datenbanken unter z/OS	353
Oracle NET8-Datenquellen	322	Dokumentation zur Funktion für Systeme zusammengesetzter Datenbanken unter Linux, UNIX und Windows	353
Sybase-Datenquellen	323	Dokumentation zu Enterprise Search unter Linux, UNIX und Windows	354
Teradata-Datenquellen	324	Release-Informationen und Installationsvoraussetzungen	355
Kapitel 29. Standardmäßige Datentypzuordnungen in Unicode	327	Eingabehilfen	357
Standardwerte für progressive Unicode-Datentypzuordnungen - NET8-Wrapper	327	Tastatureingabe und Navigation	357
Standardmäßige regressive Unicode-Datentypzuordnungen - NET8-Wrapper	327	Tastatureingabe.	357
Standardmäßige progressive Unicode-Datentypzuordnungen - Sybase-Wrapper	328	Navigation über die Tastatureingabe.	357
Standardmäßige regressive Unicode-Datentypzuordnungen - Sybase-Wrapper	329	Tastatureingabebereich	357
Standardwerte für progressive Unicode-Datentypzuordnungen - ODBC-Wrapper	329	Eingabehilfen für Bildschirme	358
Standardmäßige regressive Unicode-Datentypzuordnungen - ODBC-Wrapper	330	Schriftarteinstellungen	358
Standardmäßige progressive Unicode-Datentypzuordnungen - Microsoft SQL Server-Wrapper	330	Unabhängigkeit von Farben	358
Standardmäßige regressive Unicode-Datentypzuordnungen - Microsoft SQL Server-Wrapper	331	Kompatibilität mit Unterstützungseinrichtungen	358
Kapitel 30. Für nicht relationale Datenquellen unterstützte Datentypen	333	Dokumentation im behindertengerechten Format	358
Vom BioRS-Wrapper unterstützte Datentypen	333	Bemerkungen	359
Vom BLAST-Wrapper unterstützte Datentypen	333	Marken	361
Vom Documentum-Wrapper unterstützte Datentypen	334	Index	363
Vom Entrez-Wrapper unterstützte Datentypen	334	Kontaktaufnahme mit IBM	369
Vom Excel-Wrapper unterstützte Datentypen	335		
Vom Extended Search-Wrapper unterstützte Datentypen	335		
Vom HMMER-Wrapper unterstützte Datentypen	335		
Vom Wrapper für Dateien mit Tabellenstruktur unterstützte Datentypen.	336		
Vom Web-Services-Wrapper unterstützte Datentypen	336		
Vom WebSphere Business Integration-Wrapper unterstützte Datentypen.	337		

Produktinformationen	369	Kommentare zur Dokumentation.	369
--------------------------------	-----	---------------------------------------	-----

Zu diesem Handbuch

In diesem Handbuch wird beschrieben, wie Sie mit einem System zusammengesetzter Datenbanken arbeiten können, nachdem Sie den Server mit zusammengesetzten Datenbanken für den Zugriff auf die Datenquellen eingerichtet und konfiguriert haben.

Im vorliegenden Handbuch werden die folgenden Themen behandelt:

- Eine Einführung in die Konzepte, die Komponenten und das Leistungsspektrum von Systemen zusammengesetzter Datenbanken
- Anweisungen zur Änderung der Konfiguration des Servers mit zusammengesetzten Datenbanken und zur Änderung der Konfiguration der Datenquellen
- Eine Erläuterung der Transaktionsunterstützung in Systemen zusammengesetzter Datenbanken
- Empfehlungen für die Verwaltung und Optimierung Ihres Systems zusammengesetzter Datenbanken zur Erzielung einer optimalen Leistung
- Eine Erläuterung der Punkte, die Sie bei der Entwicklung von Anwendungen für ein System zusammengesetzter Datenbanken beachten müssen
- Detaillierte Referenzinformationen zu jeder Datenquelle

Die vertikalen Linien an den linken Rändern in diesem Buch weisen Sie darauf hin, dass am entsprechenden Text technische Änderungen vorgenommen wurden.

Zielgruppe

Dieses Buch richtet sich an Systemadministratoren, Datenbankadministratoren, Sicherheitsadministratoren und Systembediener, die ein System zusammengesetzter Datenbanken einrichten, konfigurieren, verwalten oder benutzen. Verwenden Sie dieses Buch bei der Verwaltung eines Systems zusammengesetzter Datenbanken für den Zugriff auf Daten relationaler und nicht relationaler Datenquellen. Dieses Buch kann auch von Programmierern und anderen Benutzern verwendet werden, die sich mit der Konfiguration, Administration, Anwendungsentwicklung und Verwendung eines Systems zusammengesetzter Datenbanken vertraut machen wollen.

In diesem Buch wird davon ausgegangen, dass Sie Kenntnisse über DB2 UDB besitzen. Sie sollten mit der standardmäßigen Datenbankterminologie, dem Entwurf von Datenbanken und der Datenbankverwaltung vertraut sein. In diesem Buch wird davon ausgegangen, dass Sie mit den von Ihnen verwendeten Anwendungen sowie mit den Datenquellen, auf die Sie zugreifen möchten, vertraut sind.

Teil 1. Einführung

Kapitel 1. Überblick über ein System zusammengesetzter Datenbanken

In diesem Kapitel werden die Funktionen eines Systems zusammengesetzter Datenbanken beschrieben und die verwendeten Konzepte und die Terminologie für Systeme zusammengesetzter Datenbanken definiert. Außerdem werden die Möglichkeiten aufgezeigt, wie Sie über Schnittstellen mit einem System zusammengesetzter Datenbanken interagieren können.

System zusammengesetzter Datenbanken

Bei einem DB2[®]-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[™]-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.

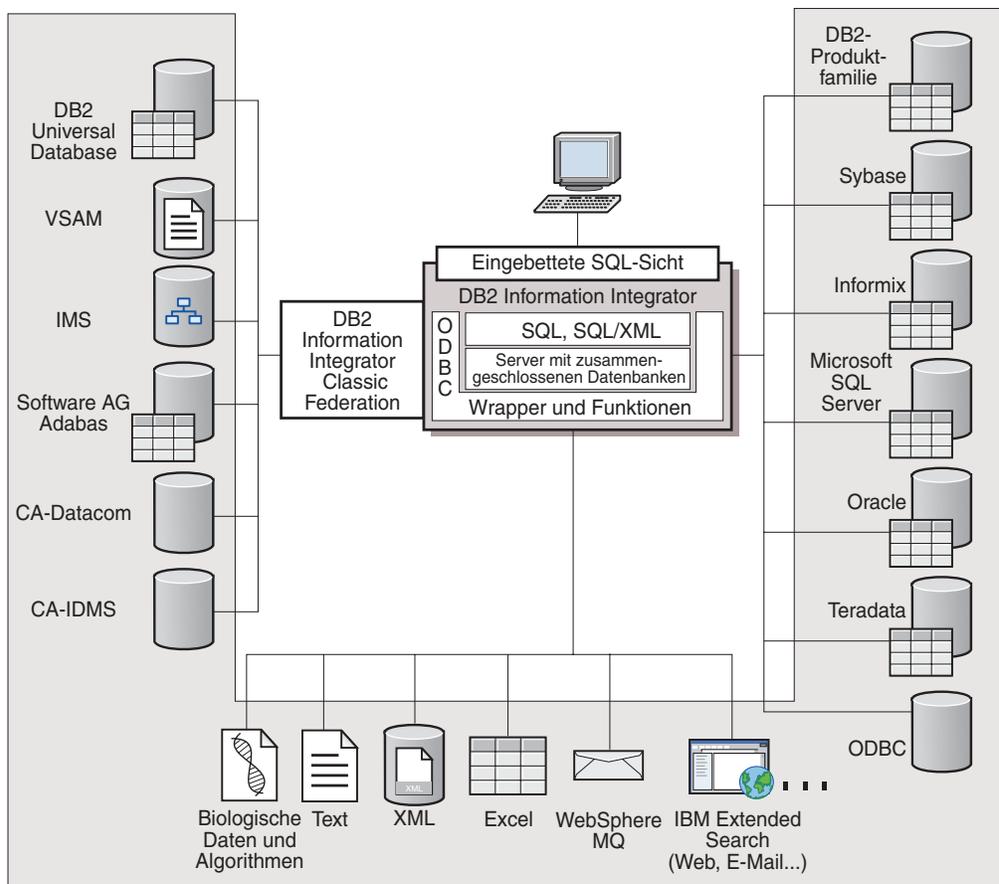


Abbildung 1. Komponenten eines Systems zusammenschlossener Datenbanken

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[®]-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[®] 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[®] Notes-Datenbanken, Microsoft[®] 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[®] für den Zugriff auf Datenquellen von DB2[®] für z/OS[™] und OS/390[®] 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"> • PQ62695 • PQ55393 • PQ56616 • PQ54605 • PQ46183 • PQ62139 8.1	DRDA
DB2 Universal Database für iSeries™	5.1 <ul style="list-style-type: none"> • mit den folgenden angewendeten APARs: <ul style="list-style-type: none"> – SE06003 – SE06872 – II13348 • mit den folgenden angewendeten PTFs: <ul style="list-style-type: none"> – SI05990 SI05991 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[®]-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:

- Kapitel 19, „Sichten in der globalen Katalogtabelle mit Informationen der zusammengeschlossenen Datenbank“, auf Seite 247

Der SQL-Compiler

Um Daten von Datenquellen abzurufen, übergeben Benutzer und Anwendungen Abfragen in DB2[®]-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[®] 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“ auf Seite 153

Kompensation

Der DB2[®]-Server mit zusammengeschlossenen Datenbanken leitet ein Abfragefragment nicht im Pushdown-Modus weiter, wenn die Datenquelle es nicht verarbeiten kann oder wenn der Server mit zusammengeschlossenen 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 zusammengeschlossenen 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 zusammengeschlossenen 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 zusammengeschlossenen 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 relation-
| alen Datenquelle, kann der Server mit zusammengeschlossenen Datenbanken die-
| ses 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 zusammengeschlossenen 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[®]-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“ auf Seite 242

Wrapper und Wrappermodule

Wrapper umfassen Mechanismen, mit deren Hilfe Server mit zusammengeschlossenen Datenbanken mit den verwendeten Datenquellen interagieren können. Der Server mit zusammengeschlossenen 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[®]-Exemplareigner verwendet normalerweise die Anweisung CREATE WRAPPER, um einen Wrapper im System zusammengeschlossener Datenbanken zu registrieren. Sie können einen Wrapper mit der Wrapperoption DB2_FENCED als abgeschirmt 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[™], eine Tabelle von DB2 für iSeries[™], zwei Tabellen von Informix[®] 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 zusammengeschlossenen Datenbank registriert sind, können Sie diese für den Zugriff auf andere Objekte in diesen Datenquellen verwenden. Sie können beispielsweise den DRDA[®]-Wrapper für alle Datenquellenobjekte der DB2-Produktfamilie, d. h. für DB2 für Linux, UNIX[®] und Windows[®], DB2 für z/OS und OS/390[®], 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 Konzepte:

- „Serverdefinitionen und Serveroptionen“ auf Seite 16

Zugehörige Tasks:

- „Trusted and fenced mode process environments“ in *IBM DB2 Information Integrator Wrapper Developer's Guide*
- „Ändern eines Wrappers“ auf Seite 31

Zugehörige Referenzen:

- „ALTER WRAPPER statement“ in *SQL Reference, Volume 2*
- Kapitel 20, „Wrapperoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 251
- „Standardnamen für Wrapper“ auf Seite 15

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
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 Exemplareigner die Datenquellen für die zusammengeschlossene Datenbank an. Der Exemplareigner 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®-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 Exemplareigner 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 Konzepte:

- „Benutzerzuordnungen“ auf Seite 17

Zugehörige Referenzen:

- Kapitel 21, „Serveroptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 253

Benutzerzuordnungen

Wenn ein Server mit zusammengeschlossenen 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 zusammengeschlossenen 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 zusammengeschlossenen Datenbanken und der Benutzer-ID der Datenquelle mit dem zugehörigen Kennwort definieren. Diese Zuordnung kann für alle Benutzer-IDs erstellt werden, die das System zusammengeschlossener 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 zusammengeschlossenen 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“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Zugehörige Referenzen:

- Kapitel 22, „Benutzerzuordnungsoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 271

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 zusammen-
| geschlossenen 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 Kurz-
| namen nicht ausgeführt werden.

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

Zugehörige Konzepte:

- „Kurznamenspaltenoptionen“ auf Seite 20

Zugehörige Referenzen:

- Kapitel 24, „Kurznamenspaltenoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 283
- Kapitel 23, „Kurznamenoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 273
- „Gültige Datenquellenobjekte“ auf Seite 19

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 zusammenschlossener 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 indexierte 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
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 20

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[®]-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 ALTER NICKNAME-Anweisungen können Sie Kurznamenspaltenoptionen
| für relationale Kurznamen definieren. Mit den Anweisungen CREATE NICKNAME
| und ALTER NICKNAME ist es möglich, Kurznamenspaltenoptionen für nicht rela-
| tionale Kurznamen anzugeben.

Zugehörige Konzepte:

- „Datentypzuordnungen“ auf Seite 21

Zugehörige Tasks:

- „Arbeiten mit Kurznamen“ auf Seite 114

Zugehörige Referenzen:

- Kapitel 24, „Kurznamenspaltenoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 283

Datentypzuordnungen

Die Datentypen der Datenquelle müssen den entsprechenden Datentypen von DB2[®] 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[™] 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[®]-Wrapper enthalten. Die Standardtypzuordnungen für Informix[®] 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[®] und Windows[®] 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“ auf Seite 53

Funktionszuordnungen

Damit der Server mit zusammengeschlossenen Datenbanken eine Datenquellenfunktion erkennen kann, muss die Funktion einer gleichwertigen Funktion in DB2[®] für Linux, UNIX[®] und Windows[®] 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“ auf Seite 67
- „Indexspezifikationen“ auf Seite 22

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“ auf Seite 83

Sortierfolgen

Die Reihenfolge, in der Zeichendaten in einer Datenbank sortiert werden, hängt von der Struktur der Daten und der Sortierfolge ab, die für die Datenbank definiert wurde.

Wenn die Daten in einer Datenbank z. B. vollständig in Großbuchstaben codiert sind und keine numerischen Werte oder Sonderzeichen enthalten, dann resultiert eine Sortieroperation unabhängig davon, ob diese in der Datenquelle oder in der zusammengeschlossenen Datenbank ausgeführt wurde, in den gleichen Ausgabedaten. Die von den einzelnen Datenbanken verwendete Sortierfolge sollte in diesem Fall keine Auswirkungen auf die Sortierergebnisse haben. Wenn alle Daten in der Datenbank in Kleinbuchstaben codiert sind und aus numerischen Zeichen bestehen, dann sollte eine Sortieroperation unabhängig vom System, auf dem diese ausgeführt wird, zu identischen Ergebnissen führen.

Setzen sich die gespeicherten Daten aus

- einer Kombination aus Buchstaben und numerischen Zeichen
- Groß- und Kleinbuchstaben
- Sonderzeichen wie z. B. @, #, €

zusammen, können Sortieroperationen in diesen Daten zu unterschiedlichen Ausgabedaten führen, wenn die zusammengeschlossene Datenbank und die Datenquelle unterschiedliche Sortierfolgen verwenden.

Allgemein kann eine *Sortierfolge* als definierte Anordnung von Zeichendaten beschrieben werden, mit der festgelegt werden kann, ob ein bestimmtes Zeichen innerhalb der Sortierung an höherer, niedrigerer oder gleicher Position wie ein anderes Zeichen eingeordnet wird.

Bestimmen der Sortierreihenfolge durch die Sortierfolge

Eine Sortierfolge legt die Sortierreihenfolge der Zeichen innerhalb eines codierten Zeichensatzes fest. Bei einem *Zeichensatz* handelt es sich um eine Gruppe von Zeichen, die in einem Datenverarbeitungssystem oder einer Programmiersprache eingesetzt wird. Innerhalb eines *codierten* Zeichensatzes ist jedem Zeichen ein bestimmter numerischer Wert innerhalb des Bereichs von 0 bis 255 (bzw. dessen hexadezimaler Entsprechung) zugeordnet. Diese numerischen Werte werden als *Codepunkte*, die Zuordnungen der numerischen Werte zu den einzelnen Zeichen der Gruppe werden zusammen als *Codepage* bezeichnet.

Zusätzlich zu seiner Zuordnung zu einem bestimmten Zeichen kann ein Codepunkt auch der Position des zugehörigen Zeichens innerhalb der Sortierreihenfolge zugeordnet sein. In der Fachsprache wird eine Sortierfolge also als die Gesamtheit der Zuordnungen der Codepunkte eines Zeichensatzes zu den Sortierreihenfolgepositionen der einzelnen Zeichen dieses Zeichensatzes bezeichnet. Die Position eines Zeichens wird durch eine Zahl dargestellt, die als *Wertigkeit* des Zeichens bezeichnet wird. In der einfachsten Sortierfolge, der so genannten *Identitätssortierfolge*, stimmen die Wertigkeiten mit den jeweiligen Codepunkten überein.

Beispiel: Die Datenbank ALPHA verwendet die Standardsortierfolge der EBCDIC-Codepage, die Datenbank BETA die der ASCII-Codepage. Die Sortierreihenfolgen für Zeichenfolgen in den beiden Datenbanken weichen wie im folgenden Beispiel definiert voneinander ab:

```
SELECT.....
ORDER BY COL2
```

EBCDIC-Sortierung	ASCII-Sortierung
COL2	COL2
----	----
V1G	7AB
Y2W	V1G
7AB	Y2W

In analoger Weise hängt die Art des Zeichenvergleichs innerhalb einer Datenbank ebenfalls von der Sortierfolge ab, die für diese Datenbank definiert wurde. In diesem Beispiel verwendet die Datenbank ALPHA die Standardsortierfolge der EBCDIC-Codepage. Die Datenbank BETA verwendet die Standardsortierfolge der ASCII-Codepage. Zeichenvergleichsoperationen in diesen beiden Datenbanken erzielen die im folgenden Beispiel aufgeführten unterschiedlichen Ergebnisse:

```
SELECT.....
WHERE COL2 > 'TT3'
```

EBCDIC-Ergebnisse	ASCII-Ergebnisse
COL2	COL2
----	----
TW4	TW4
X82	X82
39G	

Definieren der lokalen Sortierfolge zum Optimieren von Abfragen

Administratoren können zusammengesessene Datenbanken erstellen, deren Sortierfolge mit der Sortierfolge der Datenquelle übereinstimmt. In diesem Fall wird die Serveroption `COLLATING_SEQUENCE` für alle Serverdefinitionen der Datenquellen auf den Wert 'Y' gesetzt. Mit dieser Einstellung wird die zusammengesessene Datenbank darüber informiert, dass die Sortierfolgen der zusammengesessenen Datenbank und der Datenquelle übereinstimmen.

Die Sortierfolge der zusammengesessenen Datenbank wird als Teil der `CREATE DATABASE-API` definiert. Über diese API können Sie eine der folgenden Sortierungen angeben:

- Identitätssortierfolge
- Eine *System*sortierfolge (von dem zur Unterstützung der Datenbank verwendeten Betriebssystem benutzte Sortierfolge)
- Eine *angepasste* Sortierung (vordefinierte, von DB2 UDB bereitgestellte oder benutzerdefinierte Sortierung)

Angenommen, es wird eine Datenquelle von DB2 für z/OS und OS/390 verwendet. Sortieroperationen, die in einer `ORDER BY`-Klausel definiert sind, werden von einer Sortierfolge implementiert, die auf einer EBCDIC-Codepage basiert. Wenn Sie DB2 für z/OS und OS/390-Daten in einem nach der Klausel `ORDER BY` sortierten

Format abrufen wollen, müssen Sie die zusammengeschlossene Datenbank so konfigurieren, dass die vordefinierte Sortierfolge auf der Basis der entsprechenden EBCDIC-Codepage verwendet wird.

Zugehörige Konzepte:

- „Serverkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 158
- „Sortierfolgen in Systemen zusammengeschlossener Datenbanken“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Zugehörige Tasks:

- „Erstellen einer zusammengeschlossenen Datenbank“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Zugehörige Referenzen:

- „Versionen in anderen Nationalsprachen“ in *Systemverwaltung: Konzept*
- „Überlegungen zur Landessprache zusammengeschlossener Datenbanken“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Interaktion mit einem System zusammengeschlossener Datenbanken

Da es sich bei der zusammengeschlossenen Datenbank um eine Datenbank von DB2® Universal Database handelt, können Sie zur Interaktion mit einem System zusammengeschlossener 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 zusammengeschlossenen 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 zusammengeschlossener 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 zusammengesetzter 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 zusammengesetzter 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 Hilfefknopf, 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 zusammengesetzter 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 zusammengesetzten 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 zusammengesetzten 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 zusammengeschlossenen Datenbank kann auch mit Hilfe des Web-Services-Wrappers über einen Web-Service-Provider ausgeführt werden.

Zugehörige Konzepte:

- „Dokument zu Web-Services-Wrapper und der Web-Services-Datendefinitionssprache“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Zugehörige Tasks:

- „Hinzufügen von Datenquellen für Web-Services zu einem Server mit zusammengeschlossenen Datenbanken“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Teil 2. Verwaltung und Pflege

Kapitel 2. Ändern von Datenquellenkonfigurationen

In regelmäßigen Abständen sind Anpassungen an der Konfiguration erforderlich, die Sie ursprünglich für den Zugriff auf Ihre Datenquellen aufgebaut haben. Beispielsweise müssen Sie eine Serverdefinition für einen neuen Server registrieren, auf den Sie zugreifen wollen. Es ist möglich, dass Sie die Benutzerzuordnung zwischen der zusammengeschlossenen Datenbank und der fernen Datenquelle ändern müssen, wenn in der fernen Datenquelle das Kennwort geändert wird. Möglicherweise müssen Sie einem Kurznamen eine Spaltenoption hinzufügen, um die Leistung zu verbessern.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Ändern eines Wrappers“
- „Ändern von Serverdefinitionen und Serveroptionen“ auf Seite 32
- „Ändern einer Benutzerzuordnung“ auf Seite 37
- „Ändern eines Kurznamens“ auf Seite 39
- „Löschen eines Wrappers“ auf Seite 47
- „Löschen einer Serverdefinition“ auf Seite 49
- „Löschen einer Benutzerzuordnung“ auf Seite 50
- „Löschen eines Kurznamens“ auf Seite 51

Ändern eines Wrappers

Nach der Konfiguration eines Wrappers können Sie die Konfiguration auf der Basis Ihrer Systemanforderungen ändern. Verwenden Sie die Anweisung ALTER WRAPPER, um folgende Aktionen auszuführen:

- Hinzufügen, Definieren oder Löschen einer oder mehrerer Wrapperoptionen
- Definieren der Umgebungs-, Registrierdatenbank- oder der Profilvariablen

Voraussetzungen:

Die Berechtigungs-ID, die der Anweisung zugeordnet ist, muss über die Berechtigung SYSADM oder DBADM verfügen.

Einschränkungen:

Sie dürfen die Wrapperoption DB2_FENCED nicht löschen.

Vorgehensweise:

Sie können einen Wrapper von der DB2-Steuerzentrale oder der DB2-Befehlszeile aus ändern.

Gehen Sie wie folgt vor, um einen Wrapper von der DB2-Steuerzentrale aus zu ändern:

1. Erweitern Sie den Ordner **Objekte zusammengeschlossener Datenbanken**. Die Wrapperobjekte werden im Inhaltsteilfenster des Fensters der DB2-Steuerzentrale angezeigt.

2. Klicken Sie mit der rechten Maustaste auf den zu ändernden Wrapper, und klicken Sie in der Liste der Aktionen auf **Ändern**. Das Notizbuch **Wrapper ändern** wird geöffnet.
 - Nehmen Sie auf der Seite **Einstellungen** die Änderungen vor.
 - Klicken Sie auf **Variablen setzen**, um die Umgebungsvariablen der Datenquelle für den Wrapper zu setzen. Umgebungsvariablen sind nicht für alle Wrapper erforderlich.
3. Klicken Sie auf **OK**, um den Wrapper zu ändern, und schließen Sie das Notizbuch **Wrapper ändern**.

Geben Sie die Anweisung ALTER WRAPPER ein, um einen Wrapper von der DB2-Befehlszeile aus zu ändern.

Zugehörige Tasks:

- „Installieren von DB2 Information Integrator-FixPaks“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*
- „Löschen eines Wrappers“ auf Seite 47

Zugehörige Referenzen:

- „ALTER WRAPPER statement“ in *SQL Reference, Volume 2*
- „Ändern eines Wrappers - Beispiele“ auf Seite 32

Ändern eines Wrappers - Beispiele

Um die Option DB2_FENCED für den Wrapper drda in 'Y' zu ändern, geben Sie die folgende Anweisung ein:

```
ALTER WRAPPER drda OPTIONS (SET DB2_FENCED 'Y');
```

Um die Option MODULE für den Wrapper odbc in '/opt/odbc/lib/libodbc.a(odbc.so)' zu ändern, geben Sie die folgende Anweisung ein:

```
ALTER WRAPPER odbc OPTIONS (SET MODULE '/opt/odbc/lib/libodbc.a(odbc.so)');
```

Zugehörige Tasks:

- „Ändern eines Wrappers“ auf Seite 31

Ändern von Serverdefinitionen und Serveroptionen

Eine zusammengeschlossene Datenbank kann anhand einer Serverdefinition eine Datenquelle identifizieren. Die Serverdefinition enthält einen lokalen Namen und andere Informationen zum Datenquellenserver. Die Serverdefinition wird vom Wrapper verwendet, wenn SQL-Anweisungen mit Kurznamen an die zusammengeschlossene Datenbank übergeben werden. Wenn Sie eine Serverdefinition ändern wollen, verwenden Sie die Anweisung ALTER SERVER.

Ein Teil der Informationen in einer Serverdefinition wird als Serveroptionen gespeichert. Für das Ändern von Serverdefinitionen ist es wichtig, die Optionen zu kennen, die für den Server angegeben werden können. Einige Serveroptionen konfigurieren den Wrapper, andere beeinflussen die Verwendung des Wrappers durch DB2 UDB. Serveroptionen werden in den Anweisungen CREATE SERVER und ALTER SERVER als Parameter angegeben. Außerdem können bei relationalen Datenquellen Serveroptionen mit der Anweisung SET SERVER OPTION temporär gesetzt wer-

den. Diese Anweisung setzt den Wert der Serveroption in der Serverdefinition für die Dauer einer einzelnen Verbindung zur zusammengeschlossenen Datenbank außer Kraft.

In der Anweisung ALTER SERVER beziehen sich das Wort SERVER und die Parameternamen, die mit 'server' beginnen, lediglich auf Datenquellen in einem System zusammengeschlossener Datenbanken. Sie beziehen sich nicht auf den Server mit zusammengeschlossenen Datenbanken oder auf DRDA-Anwendungsserver.

Voraussetzungen:

Die Berechtigungs-ID, die die Anweisung ALTER SERVER absetzt, muss die Berechtigung SYSADM oder DBADM für die zusammengeschlossene Datenbank enthalten.

Einschränkungen:

Sie können in der Anweisung ALTER SERVER keinen Wrapper angeben, der nicht auf dem Server mit zusammengeschlossenen Datenbanken registriert wurde.

Unter den folgenden Bedingungen kann der Server mit zusammengeschlossenen Datenbanken innerhalb einer bestimmten Arbeitseinheit (UOW = Unit of Work) keine Anweisung ALTER SERVER verarbeiten:

- Die Anweisung verweist auf eine einzelne Datenquelle, und die UOW beinhaltet bereits eine der folgenden Anweisungen:
 - Eine Anweisung SELECT, die auf einen Kurznamen für eine Tabelle oder eine Sicht innerhalb der Datenquelle verweist
 - Einen geöffneten Cursor für einen Kurznamen einer Tabelle oder Sicht in der Datenquelle
 - Eine Anweisung INSERT, DELETE oder UPDATE für einen Kurznamen einer Tabelle oder Sicht in der Datenquelle
- Die Anweisung verweist auf eine Kategorie von Datenquellen (beispielsweise alle Datenquellen eines bestimmten Typs und einer bestimmten Version), und die UOW beinhaltet bereits eine der folgenden Anweisungen:
 - Eine Anweisung SELECT, die auf einen Kurznamen für eine Tabelle oder eine Sicht innerhalb einer der Datenquellen verweist
 - Einen geöffneten Cursor für einen Kurznamen einer Tabelle oder Sicht in einer der Datenquellen
 - Eine Anweisung INSERT, DELETE oder UPDATE für einen Kurznamen einer Tabelle oder Sicht in einer der Datenquellen

Ändern Sie eine Serverdefinition, wenn Folgendes zutrifft:

- Sie führen ein Upgrade auf eine neue Version der Datenquelle durch.
- Sie wollen an allen Serverdefinitionen für einen bestimmten Datenquellentyp dieselben Änderungen vornehmen.
- Sie wollen eine Serveroption in einer vorhandenen Serverdefinition ändern.

Zugehörige Tasks:

- „Ändern der Datenquellenversion in einer Serverdefinition“ auf Seite 34
- „Löschen einer Serverdefinition“ auf Seite 49

Zugehörige Referenzen:

- Kapitel 21, „Serveroptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 253

Ändern von Serverdefinitionen und Serveroptionen - ausführliche Informationen

Verwenden Sie die Anweisung ALTER SERVER, um eine Serverdefinition zu ändern.

Ändern der Datenquellenversion in einer Serverdefinition

Sie können eine vorhandene Serverdefinition ändern, um die Version der Datenquelle zu ändern, die der ferne Server verwendet.

Voraussetzungen:

Die Berechtigungs-ID, die die Anweisung ALTER SERVER absetzt, muss die Berechtigung SYSADM oder DBADM für die zusammengeschlossene Datenbank enthalten.

Vorgehensweise:

Sie können eine Serverdefinition von der DB2-Steuerzentrale oder von der Eingabeaufforderung der Befehlszeile aus ändern.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Erweitern Sie den Ordner **Objekte zusammengeschlossener Datenbanken**. Die Serverdefinitionsobjekte werden im Inhaltsteilfenster des Fensters der DB2-Steuerzentrale angezeigt.
2. Klicken Sie mit der rechten Maustaste auf die zu ändernde Serverdefinition, und klicken Sie in der Liste der Aktionen auf **Ändern**. Das Notizbuch **Serverdefinition ändern** wird geöffnet.
3. Klicken Sie auf der Seite **Server** auf den Pfeil für die Version, um eine andere Version für die Datenquelle anzugeben.
4. Klicken Sie auf **OK**, um die Serverdefinition zu ändern, und schließen Sie das Notizbuch **Serverdefinition ändern**.

Gehen Sie wie folgt vor, um diese Task von der Eingabeaufforderung der Befehlszeile aus auszuführen:

Geben Sie die Anweisung ALTER SERVER ein.

Angenommen, Sie verfügen über eine Serverdefinition für eine Microsoft SQL Server Version 6.5-Datenquelle. Der Name, den Sie dem Server in der Anweisung CREATE SERVER zugeordnet haben, lautet QLSVR_ASIA. Wenn für den Server von Microsoft SQL Server ein Upgrade auf Version 7.0 durchgeführt wird, lautet die Anweisung zum Ändern der Serverdefinition wie folgt:

```
ALTER SERVER QLSVR_ASIA VERSION 7
```

Zugehörige Tasks:

- „Ändern aller Serverdefinitionen für einen bestimmten Datenquellentyp“ auf Seite 35

Zugehörige Referenzen:

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

Ändern aller Serverdefinitionen für einen bestimmten Datenquellentyp

Sie können alle vorhandenen Serverdefinitionen für einen bestimmten Datenquellentyp mit einer einzigen Anweisung ALTER SERVER ändern. Verwenden Sie eine einzige Anweisung, wenn Sie dieselbe Änderung auf alle Serverdefinitionen desselben Typs anwenden wollen.

Voraussetzungen:

Die Berechtigungs-ID, die die Anweisung ALTER SERVER absetzt, muss die Berechtigung SYSADM oder DBADM für die zusammengeschlossene Datenbank enthalten.

Einschränkungen:

Sie können Serveroptionen für einen ganzen Datenquellentyp mit der Anweisung ALTER SERVER nur definieren oder löschen, wenn die Serveroptionen über eine vorausgegangene Anweisung ALTER SERVER hinzugefügt wurden.

Vorgehensweise:

Angenommen, im globalen Katalog sind für ihre Sybase-Datenquellen fünf Sybase-Server registriert. Jedes Mal, wenn eine Benutzer-ID zur Authentifizierung an einen dieser Sybase-Server gesendet wird, soll der Server mit zusammengeschlossenen Datenbanken die Benutzer-ID in Großbuchstaben umsetzen. Außerdem wollen Sie festlegen, wie lange der Server mit zusammengeschlossenen Datenbanken auf eine Antwort von diesen Sybase-Servern auf eine SQL-Anweisung wartet. Sie geben die Zeit in Anzahl Sekunden an.

Sie können alle fünf Serverdefinitionen gleichzeitig ändern, indem Sie eine einzige Anweisung absetzen, zum Beispiel:

```
ALTER SERVER TYPE sybase  
  OPTIONS (ADD FOLD_ID 'U', ADD TIMEOUT '600')
```

Zugehörige Tasks:

- „Ändern der Datenquellenversion in einer Serverdefinition“ auf Seite 34

Zugehörige Referenzen:

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

Verwenden von Serveroptionen in Serverdefinitionen

Es gibt allgemeine Serveroptionen und Serveroptionen für bestimmte Datenquellentypen. Sie können eine Serverdefinition ändern, um eine Serveroption hinzuzufügen oder zu ändern.

Serveroptionen werden auf Werte gesetzt, die für aufeinander folgende Verbindungen zur Datenquelle bestehen bleiben. Diese Werte werden im Systemkatalog des Systems zusammengeschlossener Datenbanken gespeichert.

Voraussetzungen:

Die Berechtigungs-ID, die die Anweisung ALTER SERVER absetzt, muss die Berechtigung SYSADM oder DBADM für die zusammengeschlossene Datenbank enthalten.

Vorgehensweise:

Sie können eine Serverdefinition von der DB2-Steuerzentrale oder von der Eingabeaufforderung der Befehlszeile aus ändern.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Erweitern Sie den Ordner **Objekte zusammengeschlossener Datenbanken**. Die Serverdefinitionsobjekte werden im Inhaltsteilfenster des Fensters der DB2-Steuerzentrale angezeigt.
2. Klicken Sie mit der rechten Maustaste auf die zu ändernde Serverdefinition, und klicken Sie in der Liste der Aktionen auf **Ändern**. Das Notizbuch **Serverdefinition ändern** wird geöffnet.
3. Wählen Sie auf der Seite **Einstellungen** die Serveroption aus, die Sie hinzufügen oder entfernen möchten.
4. Geben Sie für Optionen, die Sie hinzufügen oder ändern, den entsprechenden Optionswert an.
5. Klicken Sie auf **OK**, um die Serverdefinition zu ändern, und schließen Sie das Notizbuch **Serverdefinition ändern**.

Einige Serveroptionen sind erforderlich und können nicht gelöscht werden. Andere Serveroptionen wiederum können nicht hinzugefügt werden, wenn bereits bestimmte Serveroptionen definiert sind. Unter den Serveroptionen für Systeme zusammengeschlossener Datenbanken finden Sie eine Liste der Beschreibungen für jede Option.

Gehen Sie wie folgt vor, um diese Task von der Eingabeaufforderung der Befehlszeile aus auszuführen:

Geben Sie die Anweisung ALTER SERVER ein. Nachfolgend sind einige Beispiele für Serveroptionen aufgeführt:

- Angenommen, Sie haben eine Serverdefinition für einen Informix-Server erstellt und dabei den Servernamen INFMX01 verwendet. Sie wollen nun die Option DB2_MAXIMAL_PUSHDOWN in 'Y' ändern. Die Anweisung zum Ändern der Serverdefinition lautet:

```
ALTER SERVER INFMX01 OPTIONS (SET DB2_MAXIMAL_PUSHDOWN 'Y')
```

- Angenommen, Sie haben eine Serverdefinition für einen Oracle-Server erstellt und dabei den Servernamen ORCL99 verwendet. Sie wollen der Definition nun die Optionen FOLD_ID und FOLD_PW hinzufügen. Die Anweisung zum Ändern der Serverdefinition lautet wie folgt:

```
ALTER SERVER ORCL99 OPTIONS (ADD FOLD_ID 'U', FOLD_PW 'U')
```

- Angenommen, Sie wollen für den Zeitlimitwert einen Zeitraum in Sekunden angeben, während dem der CTLIB-Wrapper auf eine Antwort vom Sybase-Server warten soll. Zum Angeben dieses Werts verwenden Sie die Serveroption TIMEOUT. Die Anweisung zum Ändern der Serverdefinition lautet wie folgt:

```
ALTER SERVER SYBSERVER OPTIONS (ADD TIMEOUT '60')
```

Temporäres Ändern von Serveroptionen für relationale Datenquellen

Um einen Serveroptionswert temporär für eine relationale Datenquelle zu definieren, müssen Sie die Anweisung SET SERVER OPTION verwenden. Diese Anweisung setzt den Wert der Serveroption in der Serverdefinition für die Dauer einer einzelnen Verbindung zur zusammengeschlossenen Datenbank außer Kraft. Der Überschreibungswert wird nicht im globalen Katalog gespeichert.

Ein Beispiel für die Anweisung SET SERVER OPTION lautet:

```
SET SERVER OPTION PLAN_HINTS TO 'Y' FOR SERVER ORA_SERVER
```

Wenn die Anweisung SET SERVER OPTION in statischen SQL-Anweisungen verwendet wird, hat sie keine Auswirkungen auf die Serveroption IUD_APP_SVP-T_ENFORCE.

Die Hierarchie der Serveroptionseinstellungen

Wenn Sie dieselbe Serveroption mit einem Wert für einen Datenquellentyp und mit einem anderen Wert für einen bestimmten Datenquellenserver festgelegt haben, besteht eine Hierarchie zwischen diesen Einstellungen. Angenommen, die Serveroption PLAN_HINTS ist für den Datenquellentyp ORACLE auf 'Y' gesetzt. Die Serveroption PLAN_HINTS ist jedoch in der Serverdefinition für den bestimmten Oracle-Datenquellenserver PURNELL auf 'N' gesetzt. Die Einstellung für den bestimmten Datenquellenserver setzt die Einstellung für den Datenquellentyp außer Kraft. Diese Konfiguration bewirkt, dass PLAN_HINTS für alle Oracle-Datenquellenserver aktiviert ist, mit Ausnahme von PURNELL.

Zugehörige Konzepte:

- „Serverdefinitionen und Serveroptionen“ auf Seite 16

Zugehörige Referenzen:

- Kapitel 21, „Serveroptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 253
- „ALTER SERVER statement“ in *SQL Reference, Volume 2*
- „SET SERVER OPTION statement“ in *SQL Reference, Volume 2*

Ändern einer Benutzerzuordnung

Eine Benutzerzuordnung ist die Zuordnung zwischen der Berechtigungs-ID für den Server mit zusammengeschlossenen Datenbanken und der Berechtigungs-ID für die Datenquelle. Benutzerzuordnungen sind für das Versenden von verteilten Anforderungen an die Datenquelle erforderlich.

Mit der Anweisung ALTER USER MAPPING wird die Berechtigungs-ID oder das Kennwort geändert, die bzw. das in der Datenquelle als Berechtigungs-ID für einen bestimmten Server mit zusammengeschlossenen Datenbanken verwendet wird.

Voraussetzungen:

Wenn die Berechtigungs-ID, die die Anweisung absetzt, nicht mit der Berechtigungs-ID identisch ist, die der Datenquelle zugeordnet ist, muss die Berechtigungs-ID, die die Anweisung absetzt, die Berechtigung SYSADM oder DBADM für die zusammengeschlossene Datenbank enthalten.

Einschränkungen:

Wenn eine Arbeitseinheit bereits eine der folgenden Anweisungen einschließt, kann der Server mit zusammengeschlossenen Datenbanken keine Anweisung ALTER USER MAPPING in dieser Arbeitseinheit verarbeiten:

- Eine Anweisung SELECT, die auf einen Kurznamen für eine Tabelle oder eine Sicht in der Datenquelle verweist, die die Zuordnung einschließt
- Einen geöffneten Cursor für einen Kurznamen einer Tabelle oder Sicht in der Datenquelle, die die Zuordnung einschließt
- Eine Anweisung INSERT, DELETE oder UPDATE für einen Kurznamen einer Tabelle oder Sicht in der Datenquelle, die die Zuordnung einschließt

Vorgehensweise:

Sie können eine Benutzerzuordnung von der DB2-Steuerzentrale oder von der Eingabeaufforderung der Befehlszeile aus ändern.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Erweitern Sie den Ordner **Objekte zusammengeschlossener Datenbanken**. Die Benutzerzuordnungsobjekte werden im Inhaltsteilfenster des Fensters der DB2-Steuerzentrale angezeigt.
2. Klicken Sie mit der rechten Maustaste auf die zu ändernde Benutzerzuordnung, und klicken Sie in der Liste der Aktionen auf **Ändern**. Das Fenster **Benutzerzuordnung ändern** wird geöffnet.
3. Ändern Sie den Wert der Option.
4. Klicken Sie auf **OK**, um die Benutzerzuordnung zu ändern, und schließen Sie das Fenster **Benutzerzuordnung ändern**.

Gehen Sie wie folgt vor, um diese Task von der Eingabeaufforderung der Befehlszeile aus auszuführen:

Geben Sie die Anweisung ALTER USER MAPPING ein.

Beispiel: Jenny verwendet den Server mit zusammengeschlossenen Datenbanken, um eine Verbindung zu einem Sybase-Server des Namens SYBSERVER herzustellen. Sie greift auf den Server mit zusammengeschlossenen Datenbanken mit der Berechtigungs-ID *jennifer* zu. Die Berechtigungs-ID *jennifer* ist der Berechtigungs-ID *jenn* auf dem Sybase-Server zugeordnet. Die Berechtigungs-ID für Jenny wurde auf dem Sybase-Server in *jen123* geändert. Die Anweisung ALTER USER MAPPING, mit der *jennifer* zu *jen123* zugeordnet wird, lautet wie folgt:

```
ALTER USER MAPPING FOR jennifer SERVER SYBSERVER
  OPTIONS (SET REMOTE_AUTHID 'jen123')
```

Thomas verwendet den Server mit zusammengeschlossenen Datenbanken, um eine Verbindung zu einem Oracle-Server des Namens ORASERVER herzustellen. Er greift auf den Server mit zusammengeschlossenen Datenbanken mit der Berechtigungs-ID *thomas* zu. Die Berechtigungs-ID *thomas* ist der Berechtigungs-ID *tom* auf dem Oracle-Server zugeordnet. Auf dem Oracle-Server wurde das Kennwort für Thomas geändert. Sein neues Kennwort lautet *day2night*. Die Anweisung ALTER USER MAPPING, mit der *thomas* dem neuen Kennwort zugeordnet wird, lautet:

```
ALTER USER MAPPING FOR thomas SERVER ORASERVER
  OPTIONS (SET REMOTE_PASSWORD 'day2night')
```

Bei den Benutzeroptionen REMOTE_AUTHID und REMOTE_PASSWORD muss die Groß-/Kleinschreibung beachtet werden, außer Sie setzen in der Anweisung CREATE SERVER die Serveroptionen FOLD_ID und FOLD_PW auf 'U' oder 'L'.

Zugehörige Referenzen:

- „ALTER USER MAPPING statement“ in *SQL Reference, Volume 2*
- Kapitel 22, „Benutzerzuordnungsoptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 271

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“ auf Seite 213
- „Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht“ auf Seite 219

Zugehörige Tasks:

- „Ändern der Optionen von Kurznamen“ auf Seite 44
- „Ändern eines lokalen Typs für ein Datenquellenobjekt“ auf Seite 61
- „Ändern der Spaltennamen von Kurznamen“ auf Seite 43
- „Ändern der Spaltenoptionen von Kurznamen“ auf Seite 45

Zugehörige Referenzen:

- „Einschränkungen zur Änderung von Kurznamen“ auf Seite 41
- Kapitel 23, „Kurznamenoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 273
- Kapitel 24, „Kurznamenspaltenoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 283
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*

Ändern von Kurznamen - ausführliche Informationen

Sie können die im globalen Katalog gespeicherten Spaltennamen der Datenquellen ändern und Spaltenoptionen festlegen, indem Sie die Kurznamen ändern.

Einschränkungen zur Änderung von Kurznamen

Die folgenden Einschränkungen gelten, wenn Sie einen Kurznamen ändern.

Spaltennamen

Für die folgenden Datenquellen kann die Anweisung ALTER NICKNAME nicht zum Ändern von Spaltennamen verwendet werden. Sie müssen den Kurznamen löschen und ihn mit den richtigen Spaltennamen erneut erstellen.

- BLAST
- Documentum
- HMMER

Spaltenoptionen

Wenn eine der folgenden Optionen für eine Spalte definiert ist, können Sie keine weiteren Optionen zu dieser Spalte hinzufügen:

- SOAPACTIONCOLUMN
- URLCOLUMN
- PRIMARY_KEY
- FOREIGN_KEY

Für BioRS

- Wenn Sie den Elementnamen einer Spalte unter Verwendung der Option ELEMENT_NAME ändern, wird der neue Name nicht auf seine Richtigkeit hin überprüft. Wenn in einer Abfrage auf die Spalte verwiesen wird, kann eine falsche Option zu Fehlern führen.
- Wenn Sie Änderungen an der Spaltenoption IS_INDEXED vornehmen, werden diese Änderungen vom BioRS-Server nicht überprüft. Wenn in einer Abfrage auf die Spalte verwiesen wird, kann eine falsche Option zu Fehlern führen.

Datentypen

- Wenn Sie den Datentyp einer Spalte ändern, muss der neue Datentyp mit dem Datentyp der entsprechenden Datenquellenspalte oder dem Datenquellenelement kompatibel sein. Das Ändern des lokalen Datentyps in einen Datentyp, der mit dem fernen Datentyp nicht kompatibel ist, kann zu unerwarteten Fehlern führen.
- Der Wert von *local_data_type* kann nicht LONG VARCHAR, LONG VARCHARIC, DATALINK oder ein benutzerdefinierter Datentyp sein.
- Der Datentyp der Datenquelle (*data_source_data_type*) kann kein benutzerdefinierter Typ sein.
- Für einige nicht relationale Datenquellen dürfen Sie die vorhandenen lokalen Typen nicht überschreiben oder neue lokale Typen erstellen. In der Dokumentation zu den einzelnen Datenquellenwrappern finden Sie weitere Informationen zu dieser Einschränkung.
- Wenn die lokale Spezifikation für den Datentyp einer Spalte geändert wird, wird die Gültigkeit aller Statistiken (z. B. HIGH2KEY und

LOW2KEY), die für die betreffende Spalte gesammelt wurden, vom Datenbankmanager für zusammengeschlossene Datenbanken aufgehoben.

- Der lokale Typ wird für das bestimmte Datenquellenobjekt definiert, wenn mit diesem Kurznamen darauf zugegriffen wird. Dasselbe Datenquellenobjekt kann einen anderen Kurznamen haben, der die Standarddatentypzuordnung verwendet.

Indexes

Die Anweisung ALTER NICKNAME kann nicht für die Registrierung eines neuen Datenquellenindexes in der zusammengeschlossenen Datenbank verwendet werden. Verwenden Sie die Anweisung CREATE INDEX mit der Klausel SPECIFICATION ONLY, um eine Indexspezifikation zu erstellen.

Parameter LOCAL NAME und LOCAL TYPE

- Unter folgenden Bedingungen kann die Anweisung ALTER NICKNAME nicht zum Ändern der lokalen Namen oder Datentypen für die Spalten im Kurznamen verwendet werden:
 - Wenn der Kurzname in einer Sicht, einer SQL-Methode oder einer SQL-Funktion verwendet wird.
 - Wenn Sie eine informative Integritätsbedingung zum Kurznamen definieren.
- Die Klausel federated_column_options muss zum Schluss angegeben werden, wenn Sie in der Anweisung ALTER NICKNAME außerdem den Parameter LOCAL NAME, den Parameter LOCAL TYPE oder beide Parameter angeben müssen.

Kurznamen

Die Anweisung ALTER NICKNAME kann nicht zum Ändern des Namens der BioRS-Datenbank verwendet werden, auf die durch einen BioRS-Kurznamen verwiesen oder die in einem BioRS-Kurznamen verwendet wird. Wenn der Name einer BioRS-Datenbank geändert wird, müssen Sie den Kurznamen löschen und ihn erneut erstellen.

Arbeitseinheiten

Unter den folgenden Bedingungen kann der Server mit zusammengeschlossenen Datenbanken innerhalb einer bestimmten Arbeitseinheit keine Anweisung ALTER NICKNAME verarbeiten:

- Wenn der Kurzname, auf den in der Anweisung ALTER NICKNAME verwiesen wird, in derselben Arbeitseinheit über einen geöffneten Cursor verfügt.
- Wenn für den Kurznamen, auf den in der Anweisung ALTER NICKNAME verwiesen wird, in derselben Arbeitseinheit eine Anweisung INSERT, DELETE oder UPDATE abgesetzt wird.
- Wenn bei nicht relationalen Datenquellen die Anweisung ALTER NICKNAME auf einen Kurznamen verweist, auf den in derselben Arbeitseinheit mit einer Anweisung SELECT verwiesen wird.

Zugehörige Tasks:

- „Ändern der Optionen von Kurznamen“ auf Seite 44
- „Ändern eines lokalen Typs für ein Datenquellenobjekt“ auf Seite 61
- „Ändern eines Kurznamens“ auf Seite 39
- „Ändern der Spaltennamen von Kurznamen“ auf Seite 43
- „Ändern der Spaltenoptionen von Kurznamen“ auf Seite 45

Ändern der Spaltennamen von Kurznamen

Wenn Sie einen Kurznamen erstellen, werden die Spaltennamen, die dem Datenquellenobjekt zugeordnet sind, in der zusammengeschlossenen 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 39

Zugehörige Referenzen:

- „Einschränkungen zur Änderung von Kurznamen“ auf Seite 41
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*

Ändern der Optionen von Kurznamen

Kurznamenoptionen sind Parameter, die Sie für den Kurznamen angeben, wenn Sie die Anweisungen CREATE NICKNAME und ALTER NICKNAME eingeben.

Sie können Kurznamenoptionen hinzufügen, definieren oder löschen, indem Sie die Anweisung ALTER NICKNAME verwenden.

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 finden Sie im Abschnitt zu den Einschränkungen beim Ändern von Kurznamen.

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 **Einstellungen** die entsprechenden Markierungsfelder neben den Optionen aus, die Sie hinzufügen oder entfernen möchten. Eine erforderliche Option kann nicht entfernt werden.
4. Um den Wert einer Option anzugeben oder zu ändern, klicken Sie auf das Feld **Wert** der entsprechenden Option. In Abhängigkeit von der Option können Sie in der Liste entweder einen Wert auswählen, mehrere gewünschte Werte durch Anklicken auswählen, oder Sie können einen neuen Wert eingeben.
5. 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
    OPTIONS (SET option_name 'option_string_value')
```

Beispielsweise wird der Kurzname DRUGDATA1 für die Datei drugdata1.txt erstellt. Bei dieser Datei handelt es sich um eine Datei mit Tabellenstruktur. Der ursprünglich in der Anweisung CREATE NICKNAME definierte, vollständig qualifizierte Pfad lautete /user/pat/drugdata1.txt.

Geben Sie die folgende Anweisung ein, um die Kurznamenoption FILE_PATH zu ändern:

```
ALTER NICKNAME DRUGDATA1 OPTIONS (SET FILE_PATH '/usr/kelly/data/drugdata1.txt')
```

Zugehörige Tasks:

- „Ändern eines Kurznamens“ auf Seite 39

Zugehörige Referenzen:

- „Einschränkungen zur Änderung von Kurznamen“ auf Seite 41
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- Kapitel 23, „Kurznamenoptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 273

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 39

Zugehörige Referenzen:

- „Einschränkungen zur Änderung von Kurznamen“ auf Seite 41
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- Kapitel 24, „Kurznamensspaltenoptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 283

Löschen eines Wrappers

Es gibt mehrere Gründe für das Löschen eines Wrappers.

Manchmal ist zum Zugriff auf eine Datenquelle mehr als ein Wrapper vorhanden. Die Auswahl des Wrappers hängt u. U. von der Version der Datenquellenclientsoftware ab, die Sie verwenden. Sie kann auch davon abhängen, welches Betriebssystem Sie auf Ihrem Server mit zusammenschlossenen Datenbanken verwenden. Angenommen, Sie möchten auf zwei Oracle-Tabellen und eine Oracle-Sicht zugreifen. Sie verwenden Oracle Version 8, und das Betriebssystem auf Ihrem Server mit

zusammengeschlossenen Datenbanken ist Windows NT. Zu Beginn haben Sie den SQLNET-Wrapper erstellt. Da DB2 Information Integrator den SQLNET-Wrapper nicht mehr unterstützt, können Sie diesen löschen und den NET8-Wrapper erstellen.

Ein weiterer Grund zum Löschen eines Wrappers besteht, wenn Sie keinen Zugriff mehr auf die Datenquelle benötigen, der der Wrapper zugeordnet ist. Angenommen, Ihr Unternehmen muss sowohl in Informix- als auch in Microsoft SQL Server-Datenbanken auf Clientinformationen zugreifen können. Sie erstellen einen Wrapper für die Informix-Datenquelle und einen für die Microsoft SQL Server-Datenquelle. Später entscheidet Ihr Unternehmen, die gesamten Informationen von Microsoft SQL Server auf Informix zu migrieren. Dann benötigen Sie den Microsoft SQL Server-Wrapper nicht mehr und können ihn löschen.

Achtung: Das Löschen eines Wrappers kann weit reichende Folgen haben. Es wirkt sich auch auf andere Objekte aus, die Sie im Server mit zusammengesetzten Datenbanken registriert haben:

- Alle Serverdefinitionen, benutzerdefinierten Funktionszuordnungen und Datentypzuordnungen, die in einer Abhängigkeitsbeziehung zum gelöschten Wrapper stehen, werden ebenfalls gelöscht.
- Das Löschen aller von diesem Wrapper abhängigen Serverdefinitionen wirkt sich auf die Objekte aus, die ihrerseits von diesen Serverdefinitionen abhängig sind. Alle benutzerdefinierten Funktionszuordnungen und Datentypzuordnungen sowie Benutzerzuordnungen, die von den gelöschten Serverdefinitionen abhängig sind, werden ebenfalls gelöscht.
- Alle Kurznamen, die von den gelöschten Serverdefinitionen abhängig sind, werden ebenfalls gelöscht. Das Löschen aller von den Serverdefinitionen abhängigen Kurznamen wirkt sich auf die folgenden Objekte aus, die ihrerseits von diesen Kurznamen abhängig sind:
 - Alle Indexspezifikationen, die von den gelöschten Kurznamen abhängig sind, werden gelöscht.
 - Alle Sichten für zusammengesetzte Datenbanken, die von den gelöschten Kurznamen abhängig sind, werden als funktionsunfähig markiert.
 - Alle gespeicherten Abfragetabellen, die von den gelöschten Kurznamen abhängig sind. Diese werden ebenfalls gelöscht.
- Die Gültigkeit aller Anwendungen, die von den gelöschten Objekten und funktionsunfähigen Sichten abhängig sind, wird aufgehoben.

Voraussetzungen:

Zum Absetzen der Anweisung DROP WRAPPER müssen Sie über die Berechtigung SYSADM oder DBADM verfügen.

Vorgehensweise:

Verwenden Sie die Anweisung DROP, um einen Wrapper zu löschen. Mit Hilfe der folgenden Anweisung löschen Sie beispielsweise den Microsoft SQL Server-Wrapper *MSSQLODBC3*:

```
DROP WRAPPER MSSQLODBC3
```

Zugehörige Referenzen:

- „DROP statement“ in *SQL Reference, Volume 2*
- „CREATE WRAPPER statement“ in *SQL Reference, Volume 2*

Löschen einer Serverdefinition

Wird eine Serverdefinition gelöscht, wird sie aus dem globalen Katalog gelöscht. Der Löschvorgang hat jedoch keine Auswirkungen auf das Datenquellenobjekt, auf das die Serverdefinition verweist.

Wenn Sie eine Serverdefinition löschen, wirkt sich dies auch auf andere Objekte aus, die Sie auf dem Server mit zusammengeschlossenen Datenbanken registriert haben:

- Alle benutzerdefinierten Funktionszuordnungen und Datentypzuordnungen sowie Benutzerzuordnungen, die von der gelöschten Serverdefinition abhängig sind, werden ebenfalls gelöscht.
- Alle Kurznamen, die von der gelöschten Serverdefinition abhängig sind, werden ebenfalls gelöscht. Das Löschen der von dieser Serverdefinition abhängigen Kurznamen wirkt sich auf die Objekte aus, die ihrerseits von diesen Kurznamen abhängig sind.
 - Alle Indexspezifikationen, die von den gelöschten Kurznamen abhängig sind, werden gelöscht.
 - Alle Sichten für zusammengeschlossene Datenbanken, die von den gelöschten Kurznamen abhängig sind, werden als funktionsunfähig markiert.
- Die Gültigkeit aller Anwendungen, die von den gelöschten Objekten und funktionsunfähigen Sichten abhängig sind, wird aufgehoben.

Verwenden Sie die Anweisung DROP dazu, eine Serverdefinition zu löschen.

Voraussetzungen:

Sie müssen über die Berechtigung SYSADM oder DBADM verfügen, um eine Serverdefinition zu löschen.

Einschränkungen:

Unter den folgenden Bedingungen kann der Server mit zusammengeschlossenen Datenbanken innerhalb einer bestimmten Arbeitseinheit (UOW = Unit of Work) keine Anweisung DROP SERVER verarbeiten:

- Die Anweisung verweist auf eine einzelne Datenquelle, und die UOW beinhaltet bereits eine der folgenden Anweisungen:
 - Eine Anweisung SELECT, die auf einen Kurznamen für eine Tabelle oder eine Sicht innerhalb der Datenquelle verweist
 - Einen geöffneten Cursor für einen Kurznamen einer Tabelle oder Sicht in der Datenquelle
 - Eine Anweisung INSERT, DELETE oder UPDATE für einen Kurznamen einer Tabelle oder Sicht in der Datenquelle
- Die Anweisung verweist auf eine Kategorie von Datenquellen (beispielsweise alle Datenquellen eines bestimmten Typs und einer bestimmten Version), und die UOW beinhaltet eine der folgenden Anweisungen:
 - Eine Anweisung SELECT, die auf einen Kurznamen für eine Tabelle oder eine Sicht innerhalb einer der Datenquellen verweist
 - Einen geöffneten Cursor für einen Kurznamen einer Tabelle oder Sicht in einer der Datenquellen
 - Eine Anweisung INSERT, DELETE oder UPDATE für einen Kurznamen einer Tabelle oder Sicht in einer der Datenquellen

Vorgehensweise:

Wenn Sie nicht mehr auf einen Datenquellenserver zugreifen müssen, löschen Sie die Serverdefinition aus der zusammengeschlossenen Datenbank. Sie können eine Serverdefinition unter Verwendung der DB2-Steuerzentrale oder über die Anweisung DROP im DB2-Befehlszeilenprozessor löschen.

Die Syntax zum Löschen einer Serverdefinition lautet wie folgt:

```
DROP SERVER server_name
```

Dabei gilt: *server_name* gibt die Serverdefinition an, die gelöscht werden soll.

Wenn Sie einen Informix-Server definiert haben, der den Servernamen INFMX01 verwendet, lautet die Anweisung zum Löschen der Serverdefinition wie folgt:

```
DROP SERVER INFMX01
```

Zugehörige Tasks:

- „Ändern von Serverdefinitionen und Serveroptionen“ auf Seite 32

Zugehörige Referenzen:

- „DROP statement“ in *SQL Reference, Volume 2*

Löschen einer Benutzerzuordnung

Wenn ein Benutzer den Zugriff auf eine ferne Datenquelle nicht länger benötigt, löschen Sie die Zuordnung zwischen dem Server mit zusammengeschlossenen Datenbanken und dem fernen Datenquellenserver.

Wenn der Benutzer mehr als einem Datenquellenserver zugeordnet ist, müssen Sie jede Zuordnung einzeln löschen.

Voraussetzungen:

Zum Absetzen der Anweisung DROP USER MAPPING muss die Berechtigungs-ID der Anweisung DROP über die Berechtigung SYSADM oder DBADM verfügen, sofern diese Berechtigungs-ID nicht mit der Benutzer-ID der zusammengeschlossenen Datenbank identisch ist, die in der Benutzerzuordnung angegeben ist. Wenn die Berechtigungs-ID und die Benutzer-ID übereinstimmen, sind keine Berechtigungen oder Zugriffsrechte erforderlich.

Vorgehensweise:

Zum Löschen einer Benutzerzuordnung verwenden Sie die Anweisung DROP:

```
DROP USER MAPPING FOR user_ID SERVER local_server_name
```

user_ID ist die Berechtigungs-ID für den Benutzer auf dem Server mit zusammengeschlossenen Datenbanken. *local_server_name* ist der lokale Name, der zur Angabe des fernen Datenquellenservers in der Serverdefinition verwendet wird.

Zugehörige Tasks:

- „Ändern einer Benutzerzuordnung“ auf Seite 37

Zugehörige Referenzen:

- „DROP statement“ in *SQL Reference, Volume 2*

Löschen eines Kurznamens

Es gibt mehrere Gründe für das Löschen eines Kurznamens. Beispiele:

- Wenn an der zugrunde liegenden Struktur bzw. am zugrunde liegenden Inhalt des Datenquellenobjekts wesentliche Änderungen vorgenommen wurden, möchten Sie möglicherweise den Kurznamen löschen. Anschließend können Sie ihn erneut erstellen, damit die Metadaten zum Objekt im globalen Katalog aktualisiert werden.
- Wenn Sie den Namen eines Kurznamens ändern möchten, müssen Sie den Kurznamen löschen und ihn dann unter Verwendung des neuen Namens erneut erstellen.
- Wenn Sie keinen Zugriff mehr auf das zugrunde liegende Datenquellenobjekt benötigen, können Sie den Kurznamen löschen.

Wenn ein Kurzname gelöscht wird, wird er aus dem globalen Katalog auf dem Server mit zusammengeschlossenen Datenbanken gelöscht. Der Löschvorgang hat jedoch keine Auswirkungen auf das Datenquellenobjekt, auf das der Kurzname verweist.

Wenn Sie einen Kurznamen löschen, wirkt sich dies auch auf andere Objekte aus, die Sie auf dem Server mit zusammengeschlossenen Datenbanken registriert haben:

- Das Löschen von Kurznamen wirkt sich auf die Objekte aus, die von diesen Kurznamen abhängig sind:
 - Alle Indexspezifikationen, die von den gelöschten Kurznamen abhängig sind, werden gelöscht.
 - Alle Sichten für zusammengeschlossene Datenbanken, die von den gelöschten Kurznamen abhängig sind, werden als funktionsunfähig markiert.
- Die Gültigkeit aller Anwendungen, die von den gelöschten Objekten und funktionsunfähigen Sichten abhängig sind, wird aufgehoben.

Verwenden Sie die Anweisung DROP zum Löschen eines Kurznamens.

Voraussetzungen:

Der Kurzname muss im Katalog aufgelistet sein.

Die Berechtigungs-ID der Anweisung DROP muss beim Löschen von Kurznamen zumindest über eines der folgenden Zugriffsrechte verfügen:

- Berechtigung SYSADM oder DBADM
- Zugriffsrecht DROPIN auf das Schema für den Kurznamen
- Definierender Benutzer des Kurznamens, der in der Spalte DEFINER der Katalogsicht für den Kurznamen aufgezeichnet wurde
- Zugriffsrecht CONTROL für den Kurznamen

Einschränkungen:

Unter den folgenden Bedingungen kann der Server mit zusammengeschlossenen Datenbanken bei Kurznamen für relationale Datenquellen innerhalb einer bestimmten Arbeitseinheit (UOW = Unit of Work) keine Anweisung DROP NICKNAME verarbeiten:

- Wenn ein Kurzname, auf den in der Anweisung verwiesen wird, in derselben Arbeitseinheit über einen geöffneten Cursor verfügt.

- Wenn für den Kurznamen, auf den in dieser Anweisung verwiesen wird, in derselben Arbeitseinheit eine Anweisung INSERT, DELETE oder UPDATE abgesetzt wird.

Unter einer der folgenden Bedingungen kann der Server mit zusammengeschlossenen Datenbanken bei Kurznamen für nicht relationale Datenquellen innerhalb einer bestimmten Arbeitseinheit (UOW = Unit of Work) keine Anweisung DROP NICKNAME verarbeiten:

- Wenn ein Kurzname, auf den in dieser Anweisung verwiesen wird, in derselben Arbeitseinheit über einen geöffneten Cursor verfügt.
- Wenn auf einen Kurznamen, auf den in dieser Anweisung verwiesen wird, in derselben Arbeitseinheit bereits in einer Anweisung SELECT verwiesen wird.
- Wenn für den Kurznamen, auf den in dieser Anweisung verwiesen wird, in derselben Arbeitseinheit eine Anweisung INSERT, DELETE oder UPDATE abgesetzt wird.

Vorgehensweise:

Die Syntax zum Löschen eines Kurznamens lautet wie folgt:

```
DROP NICKNAME nickname
```

Dabei kennzeichnet *nickname* den zu löschenden Kurznamen.

Zugehörige Tasks:

- „Ändern eines Kurznamens“ auf Seite 39

Zugehörige Referenzen:

- „DROP statement“ in *SQL Reference, Volume 2*

Kapitel 3. Datentypzuordnungen

Die mit DB2 Information Integrator bereitgestellten Wrapper enthalten Standard-datentypzuordnungen zwischen den Datenquellen und DB2 Universal Database für Linux, UNIX und Windows.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Datentypzuordnungen in einem System zusammenschlossener Datenbanken“
- „Datentypzuordnungen und der globale Katalog zusammenschlossener Datenbanken“ auf Seite 54
- „Zeitpunkt für die Erstellung alternativer Datentypzuordnungen“ auf Seite 55
- „Datentypzuordnungen für nicht relationale Datenquellen“ auf Seite 56
- „Progressive und regressive Datentypzuordnungen“ auf Seite 56
- „Erstellen von Datentypzuordnungen“ auf Seite 57
- „Erstellen einer Datentypzuordnung für einen bestimmten Datentyp einer Datenquelle - Beispiel“ auf Seite 58
- „Erstellen einer Typzuordnung für den Datentyp und die Version einer Datenquelle - Beispiel“ auf Seite 59
- „Erstellen einer Typzuordnung für alle Datenquellenobjekte auf einem bestimmten Server - Beispiel“ auf Seite 60
- „Ändern eines lokalen Typs für ein Datenquellenobjekt“ auf Seite 61
- „Ändern eines lokalen Typs für ein Datenquellenobjekt - Beispiele“ auf Seite 63
- „Ändern langer Datentypen in VARCHAR-Datentypen“ auf Seite 64

| Datentypzuordnungen in einem System zusammenschlossener | Datenbanken

Die Datentypen in einer Datenquelle müssen den entsprechenden Datentypen von DB2[®] zugeordnet werden. Mit dieser Zuordnung kann der Server mit zusammenschlossenen Datenbanken Daten aus der Datenquelle abrufen.

Für einige Datenquellen stellt DB2 Information Integrator eine Reihe von Standard-datentypzuordnungen bereit. Für andere Datenquellen müssen Sie die Datentypzuordnungen, die Sie verwenden wollen, selbst bereitstellen. Für nicht relationale Datenquellen können keine vorhandenen Datentypzuordnungen außer Kraft gesetzt und keine neuen Zuordnungen erstellt werden.

Einige Beispiele für standardmäßige Datentypzuordnungen:

- Der Oracle-Typ FLOAT wird standardmäßig dem DB2-Typ DOUBLE zugeordnet.
- Der Oracle-Typ DATE wird standardmäßig dem DB2-Typ TIMESTAMP zugeordnet.
- Der Typ DATE von DB2 Universal Database[™] für z/OS und OS/390[®] wird standardmäßig dem DB2-Typ DATE zugeordnet.

Kurznamen, die nach der Änderung einer Zuordnung erstellt wurden, verwenden die neue Typenzuordnung. Kurznamen, die vor der Änderung der Zuordnung erstellt wurden, verwenden die standardmäßige Datentypzuordnung.

Wenn Sie die Kurznamen bereits erstellt haben, können Sie die vorhandenen Kurznamen aktualisieren. Sie haben hierfür zwei Auswahlmöglichkeiten:

- Sie können jeden Kurznamen ändern.
- Sie können jeden Kurznamen einzeln löschen und erneut erstellen.

Für die folgenden Datentypen unterstützen Server mit zusammengeschlossenen Datenbanken von DB2 keine Zuordnungen:

- Der lokale Datentyp kann nicht LONG VARCHAR, LONG VARGRAPHIC, DATALINK oder ein benutzerdefinierter Datentyp sein.
- Der ferne Datentyp kann kein benutzerdefinierter Datentyp sein.

Allerdings können Sie eine Umsetzungsfunktion verwenden, um den benutzerdefinierten Datentyp in einer Sicht in der fernen Datenquelle, die der Datenquelle entspricht, in einen integrierten oder einen Systemdatentyp umzuwandeln. Anschließend können Sie einen Kurznamen für die Sicht erstellen. Wenn Sie solche Sichten erstellen, verfügen die Sichten über keine Statistiken oder Indizes, und Sie können die Sichten nicht aktualisieren.

Zugehörige Konzepte:

- „Datentypzuordnungen und der globale Katalog zusammengeschlossener Datenbanken“ auf Seite 54
- „Zeitpunkt für die Erstellung alternativer Datentypzuordnungen“ auf Seite 55
- „Datentypzuordnungen für nicht relationale Datenquellen“ auf Seite 56
- „Progressive und regressive Datentypzuordnungen“ auf Seite 56

Zugehörige Tasks:

- „Erstellen von Datentypzuordnungen“ auf Seite 57

Zugehörige Referenzen:

- Kapitel 27, „Standardmäßige progressive Datentypzuordnungen“, auf Seite 299
- Kapitel 28, „Standardmäßige regressive Typzuordnungen“, auf Seite 315
- „Ändern langer Datentypen in VARCHAR-Datentypen“ auf Seite 64

Datentypzuordnungen und der globale Katalog zusammengeschlossener Datenbanken

Wenn Sie die Anweisung CREATE NICKNAME schreiben, müssen Sie ein Datenquellenobjekt angeben, das vom Kurznamen dargestellt wird. In den meisten Fällen definiert der Server mit zusammengeschlossenen Datenbanken für jede Spalte bzw. jedes Feld in diesem Datenquellenobjekt einen von DB2[®] unterstützten Datentyp. Bei einigen nicht relationalen Datenquellen müssen Sie den DB2-Datentyp angeben. Diese lokalen Datentypdefinitionen werden in der Katalogsicht SYSCAT.COLUMNS im globalen Katalog der zusammengeschlossenen Datenbank gespeichert.

Für relationale Datenquellen überprüft der Server mit zusammengeschlossenen Datenbanken die Wrapper und die Katalogsicht SYSCAT.TYPEMAPPINGS auf Informationen zu progressiven Datentypzuordnungen, um zu bestimmen, welcher lokale Datentyp in der Katalogsicht SYSCAT.COLUMNS gespeichert werden soll. Zuordnungen in der Katalogsicht SYSCAT.TYPEMAPPINGS haben Vorrang vor den Standardzuordnungen in den Wrappern. Wenn Sie alternative Zuordnungen zum Außer-Kraft-Setzen der Standarddatentypzuordnungen erstellen, verwendet der Server mit zusammengeschlossenen Datenbanken die alternativen Zuordnungen.

gen. Wenn für eine Spalte mehrere Zuordnungen gelten, verwendet der Server mit zusammengeschlossenen Datenbanken die zuletzt erstellten Zuordnungen.

Für nicht relationale Datenquellen überprüft der Server mit zusammengeschlossenen Datenbanken die Wrapper auf Informationen zu Datentypzuordnungen, um zu bestimmen, welcher lokale Datentyp in der Katalogsicht SYSCAT.COLUMNS gespeichert werden soll. In Abhängigkeit von der nicht relationalen Datenquelle variiert der Grad der Modifizierbarkeit der durch den Wrapper definierten Datentypen. Für einige nicht relationale Datenquellen müssen Sie keine Spalten angeben. Der Wrapper definiert die Datentypen. Für andere Datenquellen können Sie die Datentypen außer Kraft setzen. Für wieder andere Datenquellen müssen Sie die Spaltendatentypen in der Anweisung CREATE NICKNAME angeben.

Wenn Sie für relationale Datenquellen die für CREATE TABLE erforderliche transparente DDL schreiben, geben Sie in der Anweisung DB2-Datentypen an. Der Server mit zusammengeschlossenen Datenbanken sucht nach Informationen zu den regressiven Datentypzuordnungen zwischen DB2 UDB und der Datenquelle. Der Server mit zusammengeschlossenen Datenbanken überprüft den Wrapper und die Katalogsicht SYSCAT.TYPEMAPPINGS auf diese Informationen hin.

Werden Werte von einer Spalte der Datenquelle an die zusammengeschlossene Datenbank zurückgegeben, entsprechen sie vollständig dem DB2-Datentyp, dem die Spalte der Datenquelle zugeordnet ist. Handelt es sich bei dieser Zuordnung um eine Standardzuordnung, entsprechen die Werte auch vollständig dem Datenquellentyp in der Zuordnung. Beispiel: Wenn in der zusammengeschlossenen Datenbank eine Oracle-Tabelle mit einer Spalte FLOAT definiert wurde, gilt für diese Spalte automatisch die Standardzuordnung des Oracle-Datentyps FLOAT zum DB2-Datentyp DOUBLE. Die von der Spalte zurückgegebenen Werte entsprechen sowohl dem Datentyp FLOAT als auch dem Datentyp DOUBLE vollständig.

Zugehörige Konzepte:

- „Datentypzuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 53

Zeitpunkt für die Erstellung alternativer Datentypzuordnungen

Für relationale Datenquellen können Sie alternative Datentypzuordnungen erstellen.

Diese Vorgehensweise kann in folgenden Situationen sinnvoll sein:

- Um eine Standarddatentypzuordnung außer Kraft zu setzen
Für manche Wrapper können Format oder Länge der zurückgegebenen Werte geändert werden. Sie können das Format oder die Länge ändern, indem Sie den DB2[®]-Datentyp ändern, dem die Werte entsprechen müssen. Der Oracle-Datentyp DATE z. B. wird als Zeitmarke verwendet. Er gibt das Jahrhundert, das Jahr, den Monat, den Tag, die Stunde, die Minute und die Sekunde an. Der Oracle-Datentyp DATE wird standardmäßig dem DB2-Datentyp TIMESTAMP zugeordnet. Um nur die Stunde, die Minute und die Sekunde zurückzugeben, können Sie die Standarddatentypzuordnung außer Kraft setzen, sodass der Oracle-Datentyp DATE dem DB2-Datentyp TIME zugeordnet wird. Wenn die Spalten des Oracle-Datentyps DATE abgefragt werden, werden von den Oracle-Zeitmarkenwerten nur die Werte für die Zeit an den Server mit zusammengeschlossenen Datenbanken zurückgegeben.
- Wenn keine Standardzuordnung vorhanden ist

Wenn keine Standarddatentypzuordnung für einen Datentyp der Datenquelle verfügbar ist, müssen Sie für den neuen Datentyp eine Zuordnung erstellen.

Zum Definieren neuer Datentypzuordnungen verwenden Sie die Anweisung CREATE TYPE MAPPING. Die von Ihnen erstellten Zuordnungen werden in der Katalogsicht SYSCAT.TYPEMAPPINGS in der zusammengeschlossenen Datenbank gespeichert.

Zugehörige Konzepte:

- „Datentypzuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 53

Zugehörige Tasks:

- „Erstellen von Datentypzuordnungen“ auf Seite 57

Datentypzuordnungen für nicht relationale Datenquellen

Für einige nicht relationale Datenquellen sind die Datentypzuordnungen nicht in den Wrappern enthalten. In einigen Fällen müssen Sie die Informationen zu lokalen Typen in der Anweisung CREATE NICKNAME angeben.

Im folgenden Beispiel sehen Sie, wie Spaltendatentypen für einige nicht relationale Datenquellen in der Anweisung CREATE NICKNAME angegeben werden:

```
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')
```

Zugehörige Konzepte:

- „Datentypzuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 53

Zugehörige Tasks:

- „Registrieren von Kurznamen für eine Datenquelle“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Progressive und regressive Datentypzuordnungen

Bei einer *progressiven Typzuordnung* erfolgt die Zuordnung von einem fernen Datentyp zu einem vergleichbaren lokalen Datentyp. Progressive Typzuordnungen werden bei der Erstellung eines Kurznamens für ein Datenquellenobjekt verwendet. Der jeweils vergleichbare lokale Typ für die einzelnen Spalten im Datenquellenobjekt wird im globalen Katalog gespeichert.

Bei einer *regressiven Typzuordnung* erfolgt die Zuordnung von einem lokalen Datentyp zu einem vergleichbaren fernen Datentyp. Die regressive Typzuordnung wird bei transparenter DDL verwendet.

In Abb. 2 auf Seite 57 sehen Sie die progressive und die regressive Datentypzuordnung.

Datentypzuordnung

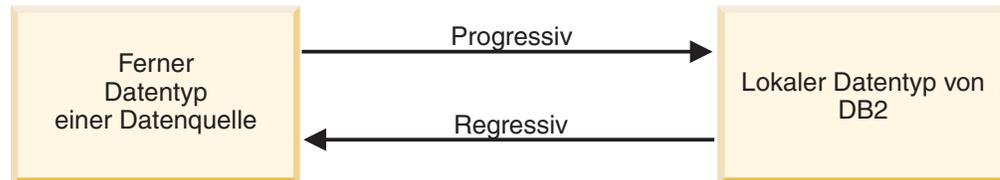


Abbildung 2. Progressive und regressive Datentypzuordnung

Zugehörige Konzepte:

- „Datentypzuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 53

Zugehörige Referenzen:

- Kapitel 27, „Standardmäßige progressive Datentypzuordnungen“, auf Seite 299
- Kapitel 28, „Standardmäßige regressive Typzuordnungen“, auf Seite 315

Erstellen von Datentypzuordnungen

Zum Erstellen einer Datentypzuordnung verwenden Sie die Anweisung CREATE TYPE MAPPING. Sie können die Anweisung von der DB2®-Befehlszentrale oder dem Befehlszeilenprozessor aus eingeben oder sie in ein Anwendungsprogramm integrieren. Die DB2-Steuerzentrale können Sie nicht zum Erstellen oder Ändern von Datentypzuordnungen verwenden.

Voraussetzungen:

Die der Anweisung zugeordneten Zugriffsrechte der Berechtigungs-ID müssen über die Berechtigung SYSADM bzw. DBADM verfügen.

Einschränkungen:

- Der Wert `local_data_type` kann nicht LONG VARCHAR, LONG VARGRAPHIC, DATALINK oder ein benutzerdefinierter Datentyp sein.
- Der Wert `data_source_data_type` kann kein benutzerdefinierter Typ sein.
- Für nicht relationale Datenquellen können keine vorhandenen Datentypzuordnungen außer Kraft gesetzt und keine Zuordnungen erstellt werden.

Vorgehensweise:

Geben Sie die Anweisung CREATE TYPE MAPPING ein, um eine Datentypzuordnung zu erstellen.

Zugehörige Konzepte:

- „Datentypzuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 53

Zugehörige Referenzen:

- Kapitel 27, „Standardmäßige progressive Datentypzuordnungen“, auf Seite 299
- Kapitel 28, „Standardmäßige regressive Typzuordnungen“, auf Seite 315

- „Erstellen einer Datentypzuordnung für einen bestimmten Datentyp einer Datenquelle - Beispiel“ auf Seite 58
- „Erstellen einer Typzuordnung für den Datentyp und die Version einer Datenquelle - Beispiel“ auf Seite 59
- „Erstellen einer Typzuordnung für alle Datenquellenobjekte auf einem bestimmten Server - Beispiel“ auf Seite 60
- „Ändern eines lokalen Typs für ein Datenquellenobjekt - Beispiele“ auf Seite 63
- „Ändern langer Datentypen in VARCHAR-Datentypen“ auf Seite 64
- „Planen der Datentypzuordnungen“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Erstellen einer Datentypzuordnung für einen bestimmten Datentyp einer Datenquelle - Beispiel

Bei diesem Beispiel müssen alle Oracle-Tabellen und -Sichten, die den Oracle-Datentyp NUMBER verwenden, dem DB2-Datentyp DECIMAL(8,2) zugeordnet werden. Der Oracle-Datentyp NUMBER wird standardmäßig dem DB2-Datentyp DOUBLE zugeordnet, einem Gleitkommadatentyp.

Verwenden Sie die Anweisung ALTER NICKNAME, um die lokalen Typen vorhandener Kurznamen zu ändern. Sie müssen jeden Kurznamen separat modifizieren, um den lokalen Datentyp in DECIMAL(8,2) zu ändern.

Wenn die Kurznamen nicht vorhanden sind, erstellen Sie eine Datentypzuordnung, die den Datenquellentyp angibt.

Wenn Sie beispielsweise eine Datentypzuordnung zwischen dem Oracle-Datentyp NUMBER und dem DB2-Datentyp DECIMAL(8,2) erstellen möchten, geben Sie die folgende Anweisung ein:

```
CREATE TYPE MAPPING MY_ORACLE_DEC FROM SYSIBM.DECIMAL (8,2)
  TO SERVER TYPE ORACLE TYPE NUMBER
```

MY_ORACLE_DEC

Der Name, den Sie der Typzuordnung geben. Der Name darf nicht mit dem Namen einer bereits im Katalog vorhandenen Datentypzuordnung identisch sein.

FROM SYSIBM.DECIMAL(8,2)

Das lokale DB2-Schema und der lokale Datentyp. Sind die Länge oder die Genauigkeit sowie der Maßstab nicht angegeben, werden diese Werte mit Hilfe des Quellendatentyps ermittelt.

TO SERVER TYPE ORACLE

Der Typ der Datenquelle.

TYPE NUMBER

Der Datentyp der Datenquelle, den Sie dem lokalen Datentyp zuordnen. Benutzerdefinierte Datentypen sind nicht zulässig.

Der DB2-Datentyp DECIMAL(8,2) ist lokal für die Oracle-Spalten definiert.

Alle neuen Oracle-Tabellen und -Sichten, die Spalten des Typs NUMBER enthalten, ordnen den Oracle-Datentyp NUMBER ebenfalls dem DB2-Datentyp DECIMAL(8,2) zu, wenn Sie die Kurznamen für diese Tabellen und Sichten erstellen.

Zugehörige Tasks:

- „Erstellen von Datentypzuordnungen“ auf Seite 57

Zugehörige Referenzen:

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

Erstellen einer Typzuordnung für den Datentyp und die Version einer Datenquelle - Beispiel

Bei diesem Beispiel sind Oracle-Tabellen und -Sichten auf Oracle-Servern unterschiedlicher Versionen vorhanden. Für alle Tabellen und Sichten, die sich auf Oracle-Servern der Version 8.0.3 befinden, müssen Spalten, die den Oracle-Datentyp NUMBER(23,3) verwenden, dem DB2-Datentyp DECIMAL(8,2) zugeordnet werden. Der Oracle-Datentyp NUMBER(23,3) wird standardmäßig dem DB2-Datentyp DECIMAL(23,3) zugeordnet.

Verwenden Sie die Anweisung ALTER NICKNAME, um die lokalen Typen vorhandener Kurznamen zu ändern. Sie müssen jeden Kurznamen separat modifizieren, um den lokalen Datentyp in DECIMAL(8,2) zu ändern.

Wenn die Kurznamen nicht vorhanden sind, erstellen Sie eine Datentypzuordnung, die den Datenquellentyp angibt.

Wenn Sie beispielsweise für Oracle-Server der Version 8.0.3 eine Datentypzuordnung zwischen dem Oracle-Datentyp NUMBER(23,3) und dem DB2-Datentyp DECIMAL(8,2) erstellen möchten, geben Sie die folgende Anweisung ein:

```
CREATE TYPE MAPPING ORA_DEC FROM SYSIBM.DECIMAL(8,2)
    TO SERVER TYPE ORACLE VERSION 8.0.3 TYPE NUMBER(23,3)
```

ORA_DEC

Der Name, den Sie der Typzuordnung geben. Der Name darf nicht mit dem Namen einer bereits im Katalog vorhandenen Datentypzuordnung identisch sein.

FROM SYSIBM.DECIMAL(8,2)

Das lokale DB2-Schema und der lokale Datentyp. Sind die Länge oder die Genauigkeit sowie der Maßstab nicht angegeben, werden diese Werte mit Hilfe des Quellendatentyps ermittelt.

TO SERVER TYPE ORACLE

Der Typ der Datenquelle.

VERSION 8.0.3

Die Version des Datenquellenservers. Sie müssen die Version angeben. Sie können auch das Release und die Releaseänderung angeben, so wie in diesem Beispiel gezeigt.

TYPE NUMBER(23,3)

Der Datentyp der Datenquelle, den Sie dem lokalen Datentyp zuordnen. Benutzerdefinierte Datentypen sind nicht zulässig.

DB2 UDB definiert den DB2-Datentyp DECIMAL(8,2) für die Oracle-Spalten auf Servern der Version 8.0.3 lokal.

Alle neuen Oracle-Tabellen und -Sichten auf Oracle-Servern der Version 8.0.3, die Spalten des Typs NUMBER(23,3) enthalten, ordnen den Oracle-Datentyp NUMBER(23,3) ebenfalls dem DB2-Datentyp DECIMAL(8,2) zu, wenn Sie die Kurznamen für diese Tabellen und Sichten erstellen.

Oracle-Tabellen und -Sichten auf Servern einer anderen Version als Version 8.0.3 verwenden die Standarddatentypzuordnung.

Zugehörige Tasks:

- „Erstellen von Datentypzuordnungen“ auf Seite 57

Zugehörige Referenzen:

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

Erstellen einer Typzuordnung für alle Datenquellenobjekte auf einem bestimmten Server - Beispiel

Bei diesem Beispiel wird der Server in der zusammengeschlossenen Datenbank als ORA2SERVER definiert. Jede Tabelle enthält eine Spalte mit dem Oracle-Datentyp DATE. Der Oracle-Datentyp DATE gibt das Jahrhundert, das Jahr, den Monat, den Tag, die Stunde, die Minute und die Sekunde an. Der Oracle-Datentyp DATE wird standardmäßig dem lokalen DB2-Datentyp TIMESTAMP zugeordnet. Wenn Sie allerdings auf diesem Server ein Objekt abfragen, darf die Ergebnismenge nur die Zeitangabe (Stunden, Minuten und Sekunden) zurückgeben.

Verwenden Sie die Anweisung ALTER NICKNAME, um die lokalen Typen vorhandener Kurznamen zu ändern. Sie müssen jeden Kurznamen separat modifizieren, um den lokalen Datentyp in TIME zu ändern.

Wenn die Kurznamen nicht vorhanden sind, erstellen Sie eine Datentypzuordnung, die den Datenquellentyp angibt.

Um den Oracle-Datentyp DATE zum DB2-Datentyp TIME für ORA2SERVER zuzuordnen, geben Sie die folgende Anweisung ein:

```
CREATE TYPE MAPPING ORA2_DATE FROM SYSIBM.TIME  
TO SERVER ORA2SERVER TYPE DATE
```

ORA2_DATE

Der Name, den Sie der Typzuordnung geben. Der Name darf nicht mit dem Namen einer bereits im Katalog vorhandenen Datentypzuordnung identisch sein.

FROM *SYSIBM.TIME*

Das lokale DB2-Schema und der lokale Datentyp. Sind die Länge oder die Genauigkeit sowie der Maßstab nicht angegeben, werden diese Werte mit Hilfe des Quellendatentyps ermittelt.

TO SERVER *ORA2SERVER*

Der lokale Name des Datenquellenservers.

TYPE DATE

Der Datentyp der Datenquelle, den Sie dem lokalen Datentyp zuordnen. Benutzerdefinierte Datentypen sind nicht zulässig.

DB2 UDB definiert den DB2-Datentyp TIME für die Oracle-Spalten des Datentyps DATE lokal.

Alle neuen Objekte, die diesem Server hinzugefügt werden und Spalten des Typs DATE enthalten, ordnen den Oracle-Datentyp DATE ebenfalls dem DB2-Datentyp TIME zu, wenn Sie die Kurznamen für diese Objekte erstellen.

Diese Datentypzuordnung beeinflusst keine Datenquellenobjekte auf anderen Oracle-Servern.

Zugehörige Tasks:

- „Erstellen von Datentypzuordnungen“ auf Seite 57

Zugehörige Referenzen:

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

Ändern eines lokalen Typs für ein Datenquellenobjekt

Wenn Sie einen Kurznamen erstellen, werden die Datentypen, die dem Datenquellenobjekt zugeordnet sind, in der zusammengeschlossenen 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 39

Zugehörige Referenzen:

- „Einschränkungen zur Änderung von Kurznamen“ auf Seite 41
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- Kapitel 27, „Standardmäßige progressive Datentypzuordnungen“, auf Seite 299
- „Ändern eines lokalen Typs für ein Datenquellenobjekt - Beispiele“ auf Seite 63
- Kapitel 30, „Für nicht relationale Datenquellen unterstützte Datentypen“, auf Seite 333

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“ auf Seite 57
- „Ändern eines lokalen Typs für ein Datenquellenobjekt“ auf Seite 61

Zugehörige Referenzen:

- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*
- „Einschränkungen zur Änderung von Kurznamen“ auf Seite 41

Ändern langer Datentypen in VARCHAR-Datentypen

Um Einfüge- und Aktualisierungsoperationen (INSERT und UPDATE) für lange Datentypen zu aktivieren, können Sie die langen Datentypen in den Datentyp VARCHAR ändern. In Tabelle 4 werden die modifizierbaren langen Datentypen nach Datenquelle sortiert aufgelistet.

Tabelle 4. Lange Datentypen, nach Datenquelle sortiert, die in den Datentyp VARCHAR geändert werden können

Datenquelle	Ferner Datentyp	Länge	Lokaler Standard datentyp	ALTER to VARCHAR
DRDA	BLOB	1 – 32672	BLOB	varchar for bit data
	CLOB	1 – 32672	CLOB	varchar
	long varchar	1 – 32672	CLOB	varchar
	long varchar for bit data	1 – 32672	BLOB	varchar for bit data
Informix	byte	1 – 32672	BLOB	varchar for bit data
	text	1 – 32672	CLOB	varchar
Microsoft SQL Server	image	1 – 32672 Hostvariablen; 1 – 8000 Literal	BLOB	varchar for bit data

Tabelle 4. Lange Datentypen, nach Datenquelle sortiert, die in den Datentyp VARCHAR geändert werden können (Forts.)

Datenquelle	Ferner Datentyp	Länge	Lokaler Standard datentyp	ALTER to VARCHAR
	text	1 – 32672 Hostvariablen; 1 – 8000 Literal	CLOB	varchar
Oracle NET8	long	1 – 32672 Hostvariablen; 1 – 4000 Literal	CLOB	varchar
	long raw	1 – 32672 Hostvariablen; 1 – 4000 Literal	BLOB	varchar for bit data
Sybase CTLIB	image	1 – 32672	BLOB	varchar for bit data
	text	1 – 32672	CLOB	varchar
Teradata	byte	32673 – 64000	BLOB	varchar for bit data(32672)
	CHAR	32673 – 64000	CLOB	varchar(32672)
	varbyte	32673 – 64000	BLOB	varchar for bit data(32672)
	varchar	32673 – 64000	CLOB	varchar(32672)

Zugehörige Konzepte:

- „Datentypzuordnungen in einem System zusammenschlossener Datenbanken“ auf Seite 53
- „Anweisungen INSERT, UPDATE und DELETE für große Objekte (LOBs)“ auf Seite 112

Zugehörige Referenzen:

- Kapitel 27, „Standardmäßige progressive Datentypzuordnungen“, auf Seite 299

Kapitel 4. Funktionszuordnungen und benutzerdefinierte Funktionen

Die mit DB2 Information Integrator bereitgestellten Wrapper enthalten Standardfunktionszuordnungen zwischen den Datenquellen und DB2 für Linux, UNIX und Windows.

- „Funktionszuordnungen in einem System zusammengeschlossener Datenbanken“
- „Funktionsweise von Funktionszuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 69
- „Voraussetzungen für die Zuordnung von benutzerdefinierten Funktionen (UDFs, User-Defined Functions)“ auf Seite 70
- „Funktionsschablonen“ auf Seite 70
- „Erstellen von Funktionsschablonen“ auf Seite 71
- „Bereitstellen von Informationen zum Funktionszuordnungsaufwand für das Abfrageoptimierungsprogramm“ auf Seite 73
- „Angaben von Funktionsnamen in einer Funktionszuordnung“ auf Seite 75
- „Erstellen von Funktionszuordnungen“ auf Seite 76
- „Benutzerdefinierte Funktionen in Anwendungen“ auf Seite 80
- „Inaktivieren einer Standardfunktionszuordnung“ auf Seite 81
- „Löschen einer benutzerdefinierten Funktionszuordnung“ auf Seite 82

Funktionszuordnungen in einem System zusammengeschlossener Datenbanken

| DB2® Information Integrator stellt Standardzuordnungen zwischen vorhandenen integrierten Datenquellenfunktionen und den entsprechenden integrierten DB2-Funktionen bereit. Damit der Server mit zusammengeschlossenen Datenbanken eine Datenquellenfunktion erkennen kann, muss die Funktion einer entsprechenden, in DB2 für Linux, UNIX® und Windows® vorhandenen Funktion zugeordnet werden.

| Die Standardfunktionszuordnungen befinden sich in den Wrappermodulen.

| Für nicht relationale Datenquellen können keine vorhandenen Funktionszuordnungen außer Kraft gesetzt und keine neuen Zuordnungen erstellt werden.

Erstellen eigener Funktionszuordnungen

Wenn keine Standardfunktionszuordnung für eine Datenquellenfunktion vorhanden ist, können Sie eine Funktionszuordnung erstellen. Ein möglicher Grund für das Fehlen einer Standardzuordnung kann darin liegen, dass in DB2 für Linux, UNIX und Windows keine entsprechende Funktion für eine Datenquellenfunktion vorhanden ist.

Wenn für eine Datenquellenfunktion zwar eine entsprechende DB2-Funktion vorhanden ist, aber trotzdem keine Standardzuordnung zur Verfügung steht, kann dies daran liegen, dass diese Funktion nicht dasselbe Ergebnis zurückgibt. Wenn die Datenquelle leicht oder deutlich abweichende Ergebnisse für bestimmte Ein-

gabedatenmengen liefert, stellen die Wrapper normalerweise keine Zuordnung zu diesen Funktionen her. Wenn Ihnen die Unterschiede in den Ergebnissen jedoch nicht so wichtig sind, können Sie eine Zuordnung zwischen den Funktionen erstellen. Das Erstellen einer Zuordnung kann zu einer Leistungssteigerung führen.

Verwenden Sie Funktionszuordnungen in folgenden Situationen:

- In der Datenquelle wird eine neue integrierte Funktion verfügbar.
- In der Datenquelle wird eine neue benutzerdefinierte Funktion verfügbar.
- Es ist keine entsprechende DB2-Funktion verfügbar.
- Es ist eine entsprechende Funktion verfügbar, die aber abweichende Ergebnisse liefert, was für Ihre Zwecke jedoch unerheblich ist.

Die Anweisungen für Funktionszuordnungen werden in der Katalogsicht SYSCAT.FUNCMAPPINGS gespeichert.

Beim Erstellen einer Funktionszuordnung kann es vorkommen, dass die Rückgabewerte einer in der Datenquelle ausgewerteten Funktion sich von den Rückgabewerten einer in der zusammengeschlossenen DB2-Datenbank ausgewerteten kompatiblen Funktion unterscheiden. DB2 Information Integrator verwendet dann die Funktionszuordnung, die jedoch zu einem SQL-Syntaxfehler oder unerwarteten Ergebnissen führen kann.

Bedeutung von Funktionszuordnungen

Funktionszuordnungen gehören zu den wichtigen Eingaben für die vom Abfrageoptimierungsprogramm ausgeführte Pushdown-Analyse. Bei der Auswahl des am besten geeigneten Abfragezugriffsplans faktorisiert das Abfrageoptimierungsprogramm die Funktionalität der Datenquelle, um einen bestimmten Typ SQL-Funktion oder -Operation auszuführen. Wenn die Funktion keine Zuordnung hat, wird die Funktion nicht zur Verarbeitung an die Datenquelle gesendet. Funktionen und andere Operationen, die im Pushdown-Modus an die Datenquelle weitergeleitet werden können, steigern die Leistung.

Wenn die Datenquelle eine einer DB2-Funktion entsprechende Funktion enthält, die jedoch leicht abweichende Ergebnisse liefert, kann durch das Erstellen einer Funktionszuordnung die Leistung gesteigert werden. Die Informix-Funktion STDEV (standard deviation = Standardabweichung) beispielsweise liefert für einige Eingabedatenmengen andere Ergebnisse als die DB2-Funktion STDDEV. Aus diesem Grund enthält der Informix-Wrapper keine Standardzuordnung zwischen diesen beiden Funktionen. Wenn für Ihre Zwecke die Abweichungen in den Ergebnissen unerheblich sind, können Sie die Leistung der Abfragen steigern, die auf Informix-Datenquellen zugreifen und die DB2-Funktion STDDEV verwenden. Indem Sie eine Zuordnung zwischen der Informix-Funktion STDEV und der DB2-Funktion STDDEV erstellen, bieten Sie dem Abfrageoptimierungsprogramm die Möglichkeit, die Verarbeitung dieser Funktion an die Datenquelle weiterzuleiten.

Zugehörige Konzepte:

- „Funktionsweise von Funktionszuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 69
- „Voraussetzungen für die Zuordnung von benutzerdefinierten Funktionen (UDFs, User-Defined Functions)“ auf Seite 70
- „Funktionsschablonen“ auf Seite 70
- „Erstellen von Funktionszuordnungen“ auf Seite 76

Zugehörige Tasks:

- „Bereitstellen von Informationen zum Funktionszuordnungsaufwand für das Abfrageoptimierungsprogramm“ auf Seite 73
- „Angaben von Funktionsnamen in einer Funktionszuordnung“ auf Seite 75

Funktionsweise von Funktionszuordnungen in einem System zusammengeschlossener Datenbanken

Wenn Sie Abfragen an den Server mit zusammengeschlossenen Datenbanken übergeben, die eine oder mehrere Funktionen beinhalten, überprüft der Server mit zusammengeschlossenen Datenbanken Informationen über die Zuordnungen zwischen DB2[®]-Funktionen und den Datenquellenfunktionen. Der Server mit zusammengeschlossenen Datenbanken überprüft diese Informationen an den folgenden beiden Positionen:

- Im Wrapper. Der Wrapper der Datenquelle enthält die Standardfunktionszuordnungen.
- In der Katalogsicht SYSCAT.FUNCMAPPINGS. Diese Sicht enthält erstellte Einträge, die die Standardfunktionszuordnungen im Wrapper überschreiben oder ergänzen. Sie enthält ebenfalls neue Zuordnungen, die Sie erstellen, wenn keine Standardfunktionszuordnung vorhanden ist. Wenn auf eine Funktion mehrere Zuordnungen angewendet werden können, wird die zuletzt erstellte angewendet.

Optionen für Funktionszuordnungen geben Informationen zu der Funktion und dem potenziellen Aufwand an, der zur Verarbeitung einer Funktion in der Datenquelle erforderlich wäre. Optionen für Funktionszuordnungen stellen beispielsweise folgende Informationen zur Verfügung:

- Name der fernen Datenquellenfunktion
- Geschätzte Anzahl der beim ersten und letzten Aufruf der Datenquellenfunktion verarbeiteten Anweisungen
- Geschätzte Anzahl der beim ersten und letzten Aufruf der Datenquellenfunktion ausgeführten Eingaben und Ausgaben
- Geschätzte Anzahl der pro Aufruf der Datenquellenfunktion verarbeiteten Anweisungen

Beim Erstellen einer Funktionszuordnung ordnen Sie eine Datenquellenfunktion einer entsprechenden Funktion in der zusammengeschlossenen Datenbank zu. Ist in DB2 keine entsprechende Funktion vorhanden, oder soll der Server mit zusammengeschlossenen Datenbanken die Datenquellenfunktion verwenden, können Sie eine Funktionsschablone als entsprechende Funktion erstellen.

Zugehörige Konzepte:

- „Funktionszuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 67
- „Voraussetzungen für die Zuordnung von benutzerdefinierten Funktionen (UDFs, User-Defined Functions)“ auf Seite 70

Voraussetzungen für die Zuordnung von benutzerdefinierten Funktionen (UDFs, User-Defined Functions)

Bevor Sie eine benutzerdefinierte Datenquellenfunktion in einem System zusammengeschlossener Datenbanken aufrufen können, muss die zusammengeschlossene Datenbank die Datenquellenfunktion einer Funktionsspezifikation zuordnen, die im globalen Katalog auf dem Server gespeichert ist.

Es gibt zwei Bedingungen, unter denen die zusammengeschlossene Datenbank eine Funktionsspezifikation einer Datenquellenfunktion zuordnen kann:

- DB2[®] UDB enthält eine Funktion, deren Kennung der Kennung der Datenquellenfunktion entspricht. Eine *Kennung* setzt sich aus dem Funktionsnamen und den Funktionseingabeparametern zusammen. Kennungen *entsprechen einander*, wenn die beiden folgenden Bedingungen erfüllt sind:
 - Sie enthalten dieselben Namen und dieselbe Anzahl von Parametern.
 - Der Datentyp jedes Parameters in der einen Kennung stimmt mit dem Datentyp des entsprechenden Parameters in der anderen Kennung überein (oder kann in diesen umgesetzt werden).
- Wenn DB2 UDB keine Funktion mit der erforderlichen Kennung enthält, können Sie eine Funktionsschablone definieren, die diese Kennung enthält. Ordnen Sie anschließend die Funktionsschablone der Datenquellenfunktion zu, die Sie aufrufen wollen.

Die Anweisungen für Funktionszuordnungen werden in der Katalogsicht SYSCAT.FUNCMAPPINGS gespeichert.

Zugehörige Konzepte:

- „Funktionszuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 67
- „Benutzerdefinierte Funktionen in Anwendungen“ auf Seite 80

Zugehörige Tasks:

- „Löschen einer benutzerdefinierten Funktionszuordnung“ auf Seite 82

Funktionsschablonen

Der Server mit zusammengeschlossenen Datenbanken erkennt eine Datenquellenfunktion, wenn eine Zuordnung zwischen der Datenquellenfunktion und einer entsprechenden DB2[®]-Funktion in der zusammengeschlossenen Datenbank vorhanden ist.

Ist keine entsprechende DB2-Funktion vorhanden, können Sie eine Funktionsschablone erstellen, die als Entsprechung fungiert.

Eine *Funktionsschablone* ist eine DB2-Funktion, die Sie erstellen, um das Aufrufen einer Datenquellenfunktion durch den Server mit zusammengeschlossenen Datenbanken zu erzwingen. Eine Funktionsschablone hat jedoch im Gegensatz zu einer regulären Funktion keinen ausführbaren Code. Empfängt der Server mit zusammengeschlossenen Datenbanken Abfragen, bei denen die Funktionsschablone angegeben wird, ruft er die Datenquellenfunktion auf.

Die Funktionsschablone wird mit der Anweisung CREATE FUNCTION unter Verwendung des Parameters AS TEMPLATE erstellt.

Nach der Erstellung einer Funktionsschablone muss die Funktionszuordnung zwischen der Schablone und der Datenquellenfunktion erstellt werden. Eine Funktionszuordnung wird mit der Anweisung CREATE FUNCTION MAPPING erstellt.

Zugehörige Konzepte:

- „Funktionszuordnungen in einem System zusammenschlossener Datenbanken“ auf Seite 67

Zugehörige Tasks:

- „Erstellen von Funktionsschablonen“ auf Seite 71

Erstellen von Funktionsschablonen

Der Server mit zusammenschlossenen Datenbanken erkennt eine Datenquellenfunktion, wenn eine Zuordnung zwischen der Datenquellenfunktion und einer entsprechenden Funktion in der zusammenschlossenen Datenbank besteht. Ist keine entsprechende Funktion vorhanden, können Sie stattdessen eine Funktionsschablone erstellen.

Voraussetzungen:

Die Berechtigungs-ID der Anweisung muss über mindestens eines der folgenden Zugriffsrechte verfügen:

- Berechtigung SYSADM oder DBADM
- Berechtigung IMPICIT_SCHEMA für die Datenbank, wenn der implizite oder der explizite Schemaname der Funktion nicht vorhanden ist
- Zugriffsrecht CREATEIN für das Schema, wenn der Schemaname der Funktion vorhanden ist

Einschränkungen:

Wenn die Datenquellenfunktion über Eingabeparameter verfügt:

- Die entsprechende Funktion in DB2 muss dieselbe Anzahl Eingabeparameter haben wie die Datenquellenfunktion.
- Die Datentypen der Eingabeparameter für die entsprechende DB2-Funktion müssen mit den zugehörigen Datentypen der Eingabeparameter für die Datenquellenfunktion kompatibel sein. Der Datentyp kann nicht LONG VARCHAR, LONG VARGRAPHIC, DATALINK oder ein benutzerdefinierter Typ sein.

Wenn die Datenquellenfunktion über keine Eingabeparameter verfügt, kann auch die entsprechende DB2-Funktion über keine Eingabeparameter verfügen.

Vorgehensweise:

Eine Funktionsschablone wird mit der Anweisung CREATE FUNCTION unter Verwendung des Parameters AS TEMPLATE erstellt. Beispiel:

```

CREATE FUNCTION BONUS ()
  RETURNS DECIMAL(8,2)
  AS TEMPLATE
  DETERMINISTIC
  NO EXTERNAL ACTION

```

BONUS ()

Der Name, den Sie der Funktionsschablone geben.

RETURNS *DECIMAL(8,2)*

Der Datentyp der Ausgabe.

AS TEMPLATE

Gibt an, dass es sich hierbei um eine Funktionsschablone und nicht um eine Funktion handelt.

DETERMINISTIC

Gibt an, dass die Funktion immer dieselben Ergebnisse für eine bestimmte Argumentwertmenge zurückgibt.

NO EXTERNAL ACTION

Gibt an, dass die Funktion keine externen Auswirkungen auf Objekte hat, die nicht vom Datenbankmanager verwaltet werden.

In Abhängigkeit davon, ob die Funktion selbst deterministisch ist und ob sie externe Aktionen auslöst, müssen Sie die Klauseln **DETERMINISTIC** und **NO EXTERNAL ACTION** angeben. Andernfalls werden den SQL-Operationen, die von dieser Funktionsschablone unterstützt werden, Einschränkungen auferlegt.

Nach der Erstellung einer Funktionsschablone muss die Funktionszuordnung zwischen der Schablone und der Datenquellenfunktion erstellt werden. Eine Funktionszuordnung wird unter Verwendung der Anweisung **CREATE FUNCTION MAPPING** erstellt. Beispiel:

```

CREATE FUNCTION MAPPING MY_INFORMIX_FUN FOR BONUS()
  SERVER TYPE INFORMIX OPTIONS (REMOTE_NAME 'BONUS()')
MY_INFORMIX_FUN

```

Der Name, den Sie der Funktionszuordnung geben. Der Name darf nicht mit dem Namen einer im DB2-Katalog bereits beschriebenen Funktionszuordnung identisch sein. Er muss eindeutig sein.

FOR *BONUS*()

Der Name der lokalen DB2-Funktionsschablone. Schließen Sie die Parameter für die Datentypangabe in runde Klammern ein.

SERVER TYPE *INFORMIX*

Gibt den Datenquellentyp an, der die Funktion enthält, zu der die Zuordnung erfolgen soll.

OPTIONS (*REMOTE_NAME* '*BONUS*()')

Eine Option für die Angabe des Namens der fernen Datenquellenfunktion, die der lokalen DB2-Funktionsschablone zugeordnet wird.

Zugehörige Konzepte:

- „Funktionszuordnungen in einem System zusammenschlossener Datenbanken“ auf Seite 67
- „Funktionsschablonen“ auf Seite 70

Zugehörige Referenzen:

- „CREATE FUNCTION (Sourced or Template) statement“ in *SQL Reference, Volume 2*

Bereitstellen von Informationen zum Funktionszuordnungsaufwand für das Abfrageoptimierungsprogramm

Wenn der DB2-SQL-Compiler eine Abfrage empfängt, die eine Funktion enthält, stellt das Abfrageoptimierungsprogramm fest, ob die Funktion im Pushdown-Modus an die Datenquelle weitergeleitet werden kann. Angenommen, die Pushdown-Analyse ergibt, dass die Funktion sowohl von der Datenquelle als auch von DB2 UDB verarbeitet werden kann.

Wenn Sie eine Funktionszuordnung erstellen, können Sie für DB2 UDB wichtige Informationen über die potenziellen Kosten bzw. den *Systemaufwand* bereitstellen, der durch Ausführen einer Datenquellenfunktion in der Datenquelle entsteht. Das Abfrageoptimierungsprogramm verwendet diese Informationen zum Vergleich des geschätzten Aufwands für die Ausführung der Datenquellenfunktion mit dem geschätzten Aufwand für die Ausführung der DB2-Funktion.

Diese Informationen unterstützen das DB2-Abfrageoptimierungsprogramm dabei, die beste Strategie für die Ausführung der betreffenden Abfrage zu ermitteln. Wenn eine verteilte Anforderung verarbeitet wird, wertet das Optimierungsprogramm mehrere Zugriffsstrategien aus und schätzt den Systemaufwand, der für den Aufruf der DB2- und der Datenquellenfunktion erforderlich ist. Die Strategie, für die der geringste Systemaufwand erwartet wird, wird verwendet.

Die geschätzten Aufwandsstatistiken sind in der Anweisung CREATE FUNCTION MAPPING anzugeben. Diese Anweisung kann beispielsweise die geschätzte Anzahl von Instruktionen angeben, die für den Aufruf der Datenquellenfunktion erforderlich sind. Sie kann die geschätzte Anzahl von Ein-/Ausgaben angeben, die für jedes Byte der Argumentgruppe aufgewendet werden muss, die an diese Funktion übergeben wird. Diese Schätzungen werden im globalen Katalog gespeichert und in der Sicht SYSCAT.FUNCMAPOPTIONS angezeigt. Wenn eine DB2-Funktion in der Zuordnung verwendet wird (an Stelle einer Datenquellenfunktion oder einer DB2-Funktionsschablone), enthält der globale Katalog Schätzungen für den Systemaufwand, der beim Aufruf der DB2-Funktion benötigt wird. Diese Schätzungen werden in der Sicht SYSCAT.ROUTINES angezeigt.

Voraussetzungen:

Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen die Berechtigung SYSADM oder DBADM umfassen.

Vorgehensweise:

Um in der Anweisung CREATE FUNCTION MAPPING geschätzte Statistiken anzugeben, müssen Sie die Optionen für Funktionszuordnungen verwenden. In der folgenden Tabelle sind die Funktionszuordnungsoptionen aufgelistet, mit denen der Funktionssystemaufwand angegeben wird, sowie die Standardwerte dieser Optionen.

Tabelle 5. Funktionszuordnungsoptionen zur Angabe des Funktionssystemaufwands

Option	Gültige Einstellungen	Standardeinstellung
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 Ein- und Ausgaben.	'0'
INSTS_PER_ARGBYTE	Geschätzte Anzahl der für jedes Byte der Argumentgruppe, die an die Datenquellenfunktion übergeben wird, verarbeiteten Anweisungen.	'0'
INSTS_PER_INVOC	Geschätzte Anzahl der pro Aufruf der Datenquellenfunktion verarbeiteten Anweisungen.	'450'
IOS_PER_ARGBYTE	Geschätzte Anzahl der für jedes Byte der Argumentgruppe, die an die Datenquellenfunktion übergeben wird, aufgewendeten Ein- und Ausgaben.	'0'
IOS_PER_INVOC	Geschätzte Anzahl von Ein- und Ausgaben pro Aufruf einer Datenquellenfunktion.	'0'
PERCENT_ARGBYTES	Geschätzter durchschnittlicher Prozentsatz der Eingabeargumentbyte, die die Datenquellenfunktion tatsächlich liest.	'100'

Beispiel: Funktionszuordnungsoption PERCENT_ARGBYTES:

Angenommen, Sie möchten eine benutzerdefinierte Funktion mit dem Namen US_DOLLAR in einer Oracle-Datenquelle zu einer benutzerdefinierten DB2-Funktion zuordnen, die Sie erstellen. Die Oracle-Datenquelle hat den Namen ORACLE2. Sie entscheiden sich dazu, die benutzerdefinierte DB2-Funktion DOLLAR und diese Funktionszuordnung ORACLE_DOLLAR zu nennen. Sie möchten die Funktionszuordnungsoption PERCENT_ARGBYTES definieren, um genauere Informationen zum Systemaufwand für das Optimierungsprogramm zur Verfügung zu stellen. Die SQL-Anweisung würde wie folgt lauten:

```
CREATE FUNCTION MAPPING ORACLE_DOLLAR FOR DOLLAR()
  SERVER ORACLE2
  OPTIONS (REMOTE_NAME 'US_DOLLAR()', PERCENT_ARGBYTES '250')
```

Beispiel: Funktionszuordnungsoption INSTS_PER_INVOC:

Angenommen, Sie wollen die lokale Funktion UCASE(CHAR) zu einer benutzerdefinierten Oracle-Funktion des Namens UPPERCASE zuordnen. Die Oracle-Funktion befindet sich in einer Datenquelle des Namens ORACLE2. Sie möchten außerdem die geschätzte Anzahl von Anweisungen pro Aufruf der benutzerdefinierten Oracle-Funktion integrieren. Die Syntax lautet wie folgt:

```
CREATE FUNCTION MAPPING MY_ORACLE_FUN4 FOR SYSFUN.UCASE(CHAR)
  SERVER ORACLE2
  OPTIONS (REMOTE_NAME 'UPPERCASE', INSTS_PER_INVOC '1000')
```

Aktualisieren der Aufwandsinformationen

Wenn sich die Schätzungen des verwendeten Systemaufwands ändern, können Sie diese Änderungen im globalen Katalog aufzeichnen. Wenn Sie neue Schätzungen für die Datenquellenfunktion aufzeichnen wollen, müssen Sie erst die Funktionszuordnung löschen oder inaktivieren. Erstellen Sie die Zuordnung anschließend erneut, und geben Sie dabei in der Anweisung CREATE FUNCTION MAPPING die neuen Schätzungen an. Die neuen Schätzungen werden zur Katalogsicht SYS-CAT.FUNCMAPPINGS hinzugefügt. Wenn Sie geänderte Schätzungen für die DB2-Funktion aufzeichnen wollen, können Sie die Katalogsicht SYSSTAT.ROUTINES direkt aktualisieren.

Zugehörige Konzepte:

- „Funktionszuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 67

Zugehörige Referenzen:

- Kapitel 25, „Funktionszuordnungsoptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 291

Angeben von Funktionsnamen in einer Funktionszuordnung

Die Werte, die Sie in der Anweisung CREATE FUNCTION MAPPING angeben, hängen davon ab, ob die Funktionen, die Sie einander zuordnen, identische oder unterschiedliche Namen haben.

Voraussetzungen:

Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen die Berechtigung SYSADM oder DBADM umfassen.

Funktionen mit identischen Namen zuordnen:

Sie können eine Zuordnung zwischen zwei Funktionen (oder zwischen einer DB2-Funktionsschablone und einer Datenquellenfunktion) mit demselben Namen erstellen. Angenommen, Sie möchten eine benutzerdefinierte Funktion mit dem Namen MYFUN in einer Informix-Datenquelle der benutzerdefinierten DB2-Funktion TINA.MYFUN zuordnen. Der Informix-Datenquellenserver hat den Namen INFORMIX2. Die Syntax dann lautet wie folgt:

```
CREATE FUNCTION MAPPING FOR TINA.MYFUN(SYSTEM.INTEGER) SERVER INFORMIX2
```

Funktionen mit unterschiedlichen Namen zuordnen:

Wenn Sie eine Zuordnung zwischen zwei Funktionen (oder zwischen einer DB2-Funktionsschablone und einer Datenquellenfunktion) erstellen, die unterschiedliche Namen haben, müssen Sie folgendermaßen vorgehen:

- Ordnen Sie den Namen der DB2-Funktion bzw. der Funktionsschablone dem Parameter *function_name* zu.
- Geben Sie eine Funktionszuordnungsoption mit dem Namen REMOTE_NAME an, und ordnen Sie den Namen der Datenquellenfunktion dieser Option zu. REMOTE_NAME muss weniger als 255 Zeichen lang sein.

Angenommen, Sie möchten eine benutzerdefinierte Funktion mit dem Namen UPPERCASE in einer Oracle-Datenquelle zur DB2-Funktion UCASE(CHAR) zuordnen. Der Oracle-Datenquellenserver hat den Namen ORACLE2. Außerdem möchten Sie die geschätzte Anzahl von Anweisungen pro Aufruf der Funktion UPPERCASE integrieren. Sie entscheiden sich dazu, diese Funktionszuordnung ORACLE_UPPER zu nennen. Die Syntax dann lautet wie folgt:

```
CREATE FUNCTION MAPPING ORACLE_UPPER FOR SYSFUN.UCASE(CHAR)
    SERVER ORACLE2 OPTIONS
    (REMOTE_NAME 'UPPERCASE', INSTS_PER_INVOC '1000')
```

Zugehörige Konzepte:

- „Funktionszuordnungen in einem System zusammengeschnittener Datenbanken“ auf Seite 67

Erstellen von Funktionszuordnungen

Verwenden Sie die Anweisung CREATE FUNCTION MAPPING, um eine alternative Funktionszuordnung anzugeben, die die Standardfunktionszuordnungen außer Kraft setzt. Wenn Sie alternative Funktionszuordnungen erstellen, werden die Einträge in der Katalogsicht SYSCAT.FUNCMAPPINGS angezeigt.

Sie können die Anweisung CREATE FUNCTION MAPPING auch zum Angeben von Funktionszuordnungsoptionen verwenden. Wenn Sie Funktionszuordnungsoptionen angeben, werden die Informationen in der Katalogsicht SYSCAT.FUNCMAPOPTIONS angezeigt.

Mit der Anweisung CREATE FUNCTION MAPPING können Sie Folgendes ausführen:

- Erstellen einer Funktionszuordnung für alle Datenquellen eines bestimmten Typs. Dies können beispielsweise alle Informix[®]-Datenquellen sein.
- Erstellen einer Funktionszuordnung für alle Datenquellen eines bestimmten Typs und einer bestimmten Version. Dies können beispielsweise alle Datenquellen von Informix 9 sein.
- Erstellen einer Funktionszuordnung für einen spezifischen Server.
- Bereitstellen von statistischen Informationen zu Funktionszuordnungen für das Optimierungsprogramm.
- Inaktivieren einer Standardfunktionszuordnung oder einer benutzerdefinierten Funktionszuordnung.

Sie können die Anweisung CREATE FUNCTION MAPPING in der DB2[®]-Befehlszentrale oder im Befehlszeilenprozessor (CLP) absetzen. Sie können die Anweisung CREATE FUNCTION MAPPING auch in ein Anwendungsprogramm einbetten. Die DB2-Steuerzentrale unterstützt das Erstellen oder Ändern von Funktionszuordnungen nicht.

Zugehörige Tasks:

- „Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp“ auf Seite 77
- „Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp und eine bestimmte Datenquellentypversion“ auf Seite 78
- „Erstellen einer Funktionszuordnung für alle Datenquellenobjekte auf einem bestimmten Server“ auf Seite 79

Erstellen von Funktionszuordnungen - ausführliche Informationen

Mit der Anweisung CREATE FUNCTION MAPPING können Sie eine Zuordnung zu einer DB2-Funktion oder einer -Funktionsschablone erstellen.

Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp

Sie können eine Funktionszuordnung für alle Datenquellen eines bestimmten Typs erstellen.

Voraussetzungen:

Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen die Berechtigung SYSADM oder DBADM umfassen.

Einschränkungen:

Für nicht relationale Datenquellen können Sie keine neuen Funktionszuordnungen erstellen, und Sie können die vorhandenen Funktionszuordnungen nicht außer Kraft setzen.

Vorgehensweise:

Angenommen, Sie wollen eine DB2-Funktionsschablone einer benutzerdefinierten Oracle-Funktion zuordnen, und zwar für alle Oracle-Datenquellen. Die Schablone hat den Namen STATS und gehört zu einem Schema mit dem Namen NOVA. Die benutzerdefinierte Oracle-Funktion hat den Namen STATISTICS und gehört zu einem Schema mit dem Namen STAR. Die Anweisung CREATE FUNCTION MAPPING lautet:

```
CREATE FUNCTION MAPPING MY_ORACLE_FUN1
  FOR NOVA.STATS ( DOUBLE, DOUBLE )
  SERVER TYPE ORACLE
  OPTIONS (REMOTE_NAME 'STAR.STATISTICS')
```

Zugehörige Konzepte:

- „Erstellen von Funktionszuordnungen“ auf Seite 76

Zugehörige Tasks:

- „Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp und eine bestimmte Datenquellentypversion“ auf Seite 78
- „Erstellen einer Funktionszuordnung für alle Datenquellenobjekte auf einem bestimmten Server“ auf Seite 79
- „Löschen einer benutzerdefinierten Funktionszuordnung“ auf Seite 82
- „Angabe von Funktionsnamen in einer Funktionszuordnung“ auf Seite 75

Zugehörige Referenzen:

- „CREATE FUNCTION MAPPING statement“ in *SQL Reference, Volume 2*

Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp und eine bestimmte Datenquellentypversion

Sie können eine Funktionszuordnung für alle Datenquellen erstellen, die eine bestimmte Version des Datenquellentyps verwenden.

Voraussetzungen:

Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen die Berechtigung SYSADM oder DBADM umfassen.

Einschränkungen:

Für nicht relationale Datenquellen können Sie keine neuen Funktionszuordnungen erstellen, und Sie können die vorhandenen Funktionszuordnungen nicht außer Kraft setzen.

Vorgehensweise:

Angenommen, Sie wollen eine DB2-Funktionsschablone einer benutzerdefinierten Sybase-Funktion zuordnen, und zwar für alle Sybase-Datenquellen, die Version 12 verwenden. Die Schablone hat den Namen SYB_STATS und gehört zu einem Schema mit dem Namen EARTH. Die benutzerdefinierte Sybase-Funktion hat den Namen STATISTICS und gehört zu einem Schema mit dem Namen MOON. Die Anweisung CREATE FUNCTION MAPPING lautet wie folgt:

```
CREATE FUNCTION MAPPING SYBASE_STATS
  FOR EARTH.SYB_STATS ( DOUBLE, DOUBLE )
  SERVER TYPE SYBASE VERSION 12
  OPTIONS (REMOTE_NAME 'MOON.STATISTICS')
```

Zugehörige Konzepte:

- „Erstellen von Funktionszuordnungen“ auf Seite 76

Zugehörige Tasks:

- „Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp“ auf Seite 77
- „Erstellen einer Funktionszuordnung für alle Datenquellenobjekte auf einem bestimmten Server“ auf Seite 79
- „Löschen einer benutzerdefinierten Funktionszuordnung“ auf Seite 82
- „Angabe von Funktionsnamen in einer Funktionszuordnung“ auf Seite 75

Zugehörige Referenzen:

- „CREATE FUNCTION MAPPING statement“ in *SQL Reference, Volume 2*

Erstellen einer Funktionszuordnung für alle Datenquellenobjekte auf einem bestimmten Server

Sie können eine Funktionszuordnung für alle Datenquellenobjekte erstellen, die sich auf einem bestimmten fernen Server befinden.

Voraussetzungen:

Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen die Berechtigung SYSADM oder DBADM umfassen.

Einschränkungen:

Für nicht relationale Datenquellen können Sie keine neuen Funktionszuordnungen erstellen, und Sie können die vorhandenen Funktionszuordnungen nicht außer Kraft setzen.

Vorgehensweise:

Angenommen, Sie möchten eine Funktionsschablone mit dem Namen BONUS zu einer benutzerdefinierten Funktion zuordnen, die ebenfalls den Namen BONUS hat. Sie wollen, dass die Funktion nur auf den Oracle-Datenserver ORA_SALES angewendet wird. Da die Funktionsnamen identisch sind, müssen Sie die Funktionszuordnungsoption REMOTE_NAME nicht angeben.

```
CREATE FUNCTION MAPPING BONUS_CALC FOR BONUS()  
SERVER ORA_SALES
```

Zugehörige Konzepte:

- „Erstellen von Funktionszuordnungen“ auf Seite 76

Zugehörige Tasks:

- „Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp“ auf Seite 77
- „Erstellen einer Funktionszuordnung für einen bestimmten Datenquellentyp und eine bestimmte Datenquellentypversion“ auf Seite 78
- „Löschen einer benutzerdefinierten Funktionszuordnung“ auf Seite 82
- „Angabe von Funktionsnamen in einer Funktionszuordnung“ auf Seite 75

Zugehörige Referenzen:

- „CREATE FUNCTION MAPPING statement“ in *SQL Reference, Volume 2*

Benutzerdefinierte Funktionen in Anwendungen

Anwendungsentwickler müssen häufig eine eigene, speziell auf ihre Anwendung oder ihre Domäne abgestimmte Funktionsgruppe erstellen. Dabei können sie benutzerdefinierte Skalarfunktionen verwenden.

Ein Einzelhandelsunternehmen könnte beispielsweise für die Überwachung der Preise der verkauften Waren einen Datentyp PRICE definieren. Möglicherweise möchte dieses Unternehmen auch eine Funktion SALES_TAX definieren. Diese Funktion würde einen gegebenen Wert für einen Preis als Eingabe verwenden, die entsprechende Verkaufssteuer berechnen und diese Daten an den anfordernden Benutzer oder die anfordernde Anwendung zurückgeben.

Diese Funktionen sind auf alle Datenbanktypen anwendbar, einschließlich LOB-Datentypen und einzigartige Datentypen. Benutzerdefinierte Funktionen erlauben in Abfragen leistungsfähige Berechnungs- und Suchvergleichselemente für die Filterung von irrelevanten Daten in der Umgebung der Datenquelle, wodurch die Antwortzeit reduziert wird. Das SQL-Optimierungsprogramm behandelt benutzerdefinierte Funktionen genauso wie integrierte Funktionen, wie z. B. SUBSTR und LENGTH. Anwendungen können in unterschiedlichen Umgebungen für Anwendungssprachen wie z. B. C, C++, COBOL und FORTRAN entwickelt werden. Die Anwendungen können eine Reihe benutzerdefinierter SQL-Funktionen gemeinsam benutzen, obwohl sie in unterschiedlichen Umgebungen für Anwendungssprachen entwickelt wurden.

Benutzerdefinierte Funktionen können Daten bearbeiten und Aktionen ausführen. Eine benutzerdefinierte Funktion kann z. B. für das Versenden einer elektronischen Nachricht oder das Aktualisieren einer Flachdatei aktiviert werden.

In DB2® können benutzerdefinierte Funktionen folgende Funktionen einschließen:

- Neu definierte Funktionen
- Im Schema SYSFUN enthaltene Funktionen. Die Beispiele enthalten mathematische Funktionen wie SIN, COS und TAN, ; wissenschaftliche Funktionen wie RADIANS, LOG10 und POWER sowie allgemeine Funktionen wie LEFT, DIFFERENCE und UCASE.

Zugehörige Konzepte:

- „User-defined functions“ in *SQL Reference, Volume 1*
- „Funktionszuordnungen in einem System zusammengeschlossener Datenbanken“ auf Seite 67
- „Voraussetzungen für die Zuordnung von benutzerdefinierten Funktionen (UDFs, User-Defined Functions)“ auf Seite 70
- „Erstellen von Funktionszuordnungen“ auf Seite 76

Zugehörige Tasks:

- „Löschen einer benutzerdefinierten Funktionszuordnung“ auf Seite 82

Inaktivieren einer Standardfunktionszuordnung

Standardfunktionszuordnungen können nicht gelöscht werden. Sie können diese Zuordnungen jedoch "funktionsunfähig" machen, indem Sie sie inaktivieren.

Voraussetzungen:

Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen die Berechtigung SYSADM oder DBADM umfassen.

Vorgehensweise:

Beim Inaktivieren einer Standardfunktionszuordnung gibt die Anweisung CREATE FUNCTION MAPPING den Namen der DB2-Funktion an und setzt die Option DISABLE auf 'Y'.

Angenommen, es gibt eine Standardfunktionszuordnung zwischen der DB2-Funktion WEEK und einer ähnlichen Funktion für Oracle-Datenquellen. Wenn eine Abfrage verarbeitet wird, die Oracle-Daten anfordert und auf WEEK verweist, können beide Funktionen aufgerufen werden. Welche Funktion aufgerufen wird, hängt davon ab, für welche Funktion die Schätzung des Abfrageoptimierungsprogramms einen geringeren Bedarf an Systemaufwand ergibt.

Angenommen, Sie möchten ermitteln, wie die Leistung beeinflusst wird, wenn nur die Funktion WEEK für die Datenquelle aufgerufen wird. Um sicherzustellen, dass für die Oracle-Datenquelle jedes Mal die Funktion WEEK aufgerufen wird, müssen Sie die Standardfunktionszuordnung außer Kraft setzen. Die Syntax lautet wie folgt:

```
CREATE FUNCTION MAPPING FOR SYSFUN.WEEK(INT)
TYPE ORACLE OPTIONS (DISABLE 'Y')
```

Zugehörige Konzepte:

- „Funktionszuordnungen in einem System zusammenschlossener Datenbanken“ auf Seite 67

Zugehörige Tasks:

- „Löschen einer benutzerdefinierten Funktionszuordnung“ auf Seite 82

Zugehörige Referenzen:

- „CREATE FUNCTION MAPPING statement“ in *SQL Reference, Volume 2*

Löschen einer benutzerdefinierten Funktionszuordnung

Wenn Sie eine selbst erstellte Funktionszuordnung nicht mehr benötigen, können Sie diese Funktionszuordnung löschen.

Wenn Sie eine benutzerdefinierte Funktionszuordnung löschen, mit der eine Standardfunktionszuordnung außer Kraft gesetzt wurde, wird wieder die Standardfunktionszuordnung verwendet.

Benutzerdefinierte Funktionszuordnungen sind in der Sicht SYSCAT.FUNCMAPPINGS aufgelistet.

Voraussetzungen:

Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen die Berechtigung SYSADM oder DBADM umfassen.

Vorgehensweise:

Verwenden Sie zum Löschen einer selbst erstellten Funktionszuordnung die Anweisung DROP FUNCTION MAPPING.

Angenommen, Sie haben eine selbst erstellte Funktionszuordnung mit dem Namen BONUS_CALC. Zum Löschen dieser Funktionszuordnung lautet die Anweisung DROP FUNCTION MAPPING wie folgt:

```
DROP FUNCTION MAPPING BONUS_CALC
```

Zugehörige Konzepte:

- „Erstellen von Funktionszuordnungen“ auf Seite 76

Zugehörige Tasks:

- „Inaktivieren einer Standardfunktionszuordnung“ auf Seite 81

Zugehörige Referenzen:

- „DROP statement“ in *SQL Reference, Volume 2*

Kapitel 5. Indexspezifikationen

Eine *Indexspezifikation* ist ein Metadatensatz, der dem globalen Katalog hinzugefügt wird, wenn Sie für ein Datenquellenobjekt einen Kurznamen erstellen. Diese Informationen werden vom Abfrageoptimierungsprogramm zur Beschleunigung der Verarbeitung von verteilten Anforderungen verwendet. In einigen Situationen wird keine Indexspezifikation erstellt, wenn Sie für ein Datenquellenobjekt einen Kurznamen erstellen. In diesen Fällen müssen Sie die Indexspezifikation selbst erstellen.

In diesem Kapitel erhalten Sie Informationen zu folgenden Themen:

- „Indexspezifikationen in einem System zusammengeschlossener Datenbanken“
- „Erstellen von Indexspezifikationen für Datenquellenobjekte“ auf Seite 84
- „Erstellen von Indexspezifikationen für Tabellen, die neue Indizes erhalten“ auf Seite 86
- „Erstellen von Indexspezifikationen für Sichten“ auf Seite 88
- „Erstellen von Indexspezifikationen für Informix-Synonyme“ auf Seite 89

Indexspezifikationen in einem System zusammengeschlossener Datenbanken

In einem System zusammengeschlossener Datenbanken verwenden Sie die Anweisung CREATE INDEX mit einem Kurznamen, um dem globalen Katalog Indexspezifikationsinformationen zur Verfügung zu stellen. Erhält eine Tabelle einen neuen Index, verweist die Anweisung CREATE INDEX auf den Kurznamen für die Tabelle. Außerdem enthält die Anweisung Informationen über den Index der Tabelle der Datenquelle. Wird ein Kurzname für eine Sicht erstellt, verweist die Anweisung CREATE INDEX auf den Kurznamen für die Sicht. Außerdem enthält die Anweisung Informationen über den Index der der Sicht zugrunde liegenden Tabelle. Die Indexspezifikation teilt dem Server mit zusammengeschlossenen Datenbanken Informationen über die Spalten und deren Eindeutigkeitsmerkmale mit, auf deren Basis ein ferner Index erstellt wird. Informationen in den Statistiken des Indexes (z. B. zur Anzahl der eindeutigen Werte des Indexschlüssels) werden in dieser Spezifikation hingegen nicht an den Server mit zusammengeschlossenen Datenbanken übergeben.

Sie müssen keine Indexspezifikationen angeben, wenn der ferne Index zum Zeitpunkt der Erstellung des Kurznamens bereits definiert war.

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 betreffende(n) Tabellenspalte(n).

Verwenden Sie Indexspezifikationen mit relationalen Datenquellen. Bei nicht relationalen Datenquellen bewirkt das Erstellen einer Indexspezifikation keine Leistungssteigerung.

Zugehörige Tasks:

- „Erstellen von Indexspezifikationen für Datenquellenobjekte“ auf Seite 84
- „Erstellen von Indexspezifikationen für Tabellen, die neue Indizes erhalten“ auf Seite 86
- „Erstellen von Indexspezifikationen für Sichten“ auf Seite 88
- „Erstellen von Indexspezifikationen für Informix-Synonyme“ auf Seite 89
- „Kurznamenkenndaten mit Auswirkungen auf die globale Optimierung“ auf Seite 173

Erstellen von Indexspezifikationen für Datenquellenobjekte

Wenn für eine Datenquellentabelle ein Kurzname erstellt wird, stellt der Server mit zusammengeschlossenen Datenbanken für den globalen Katalog Informationen über alle Indizes der Datenquellentabelle zur Verfügung. Das Optimierungsprogramm verwendet diese Informationen zur Beschleunigung der Verarbeitung von verteilten Anforderungen. Diese Informationen bestehen aus einem Satz von Metadaten und werden als *Indexspezifikation* bezeichnet.

Der Server mit zusammengeschlossenen Datenbanken erstellt in folgenden Fällen keine Indexspezifikationen:

- Ein Kurzname wird für eine Tabelle erstellt, die keinen Index hat.
- Ein Kurzname wird für ein Datenquellenobjekt erstellt, das keine Indizes enthält. Solche Datenquellenobjekte sind beispielsweise Sichten, Informix-Synonyme, Dateien mit Tabellenstruktur, Excel-Arbeitsblätter, BLAST-Algorithmen und XML-Dateien.
- Der ferne Index besteht für eine LOB-Spalte.
- Der ferne Index enthält eine Gesamtschlüssellänge von mehr als 1024 Byte.
- Die maximale Anzahl an Schlüsselkomponenten beträgt mehr als 16.

Unter diesen Umständen speichert der Server mit zusammengeschlossenen Datenbanken keine Indexspezifikationen für die Datenquellenobjekte. Allerdings können Sie dem globalen Katalog die erforderlichen Indexinformationen für die ersten beiden Einträge in der vorherigen Liste zur Verfügung stellen. Mit Hilfe der Anweisung CREATE INDEX können Sie die Indexinformationen angeben.

Voraussetzungen:

Die Berechtigungs-ID der Anweisung muss zumindest über eines der folgenden Zugriffsrechte verfügen:

- Berechtigung SYSADM oder DBADM
- Entweder das Zugriffsrecht CONTROL oder das Zugriffsrecht INDEX für das Objekt. Außerdem muss die ID entweder über die Berechtigung IMPLICIT_SCHEMA für die Datenbank verfügen, wenn der implizite oder der explizite Schemaname des Indexes nicht vorhanden ist, oder über das Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname des Indexes auf ein bestehendes Schema bezieht.

Einschränkungen:

Bei der Erstellung einer Indexspezifikation für einen Kurznamen gibt es einige Einschränkungen.

- Wenn die Bindeoption DYNAMICRULES BIND zutrifft, kann die Anweisung nicht dynamisch vorbereitet werden. Außerdem können Sie die Parameter INCLUDE, CLUSTER, PCTFREE, MINPCTUSED, DISALLOW REVERSE SCANS und ALLOW REVERSE SCANS in der Anweisung CREATE INDEX nicht verwenden.
- UNIQUE sollte nur dann angegeben werden, wenn die Daten für den Indexschlüssel für jede Zeile der Datenquellentabelle eindeutige Werte enthalten. Die Eindeutigkeit wird nicht überprüft.
- Die Summe der gespeicherten Längen der angegebenen Spalten darf nicht größer als 1024 sein.
- Keine LOB- oder DATALINK-Spalte bzw. keine Spalte eines einzigartigen Datentyps auf der Basis von LOB oder DATALINK kann für einen Index verwendet werden. Diese Einschränkung wird auch dann durchgesetzt, wenn das Längensattribut der Spalte klein genug ist, damit die Begrenzung von 1024 Byte nicht überschritten wird.

Syntax von CREATE INDEX:

Die Anweisung CREATE INDEX kann in ein Anwendungsprogramm eingebettet oder über dynamische SQL-Anweisungen in der Steuerzentrale oder der Befehlszeile ausgegeben werden.

Wenn die Anweisung CREATE INDEX mit Kurznamen verwendet wird, erstellt sie im globalen Katalog der zusammengeschlossenen Datenbanken eine Indexspezifikation. Für die Datenquellentabelle erstellt sie keinen Index.

Verwenden Sie die folgende Syntax, um eine Indexspezifikation zu erstellen:

```
CREATE INDEX index_name ON nickname  
(column_name) SPECIFICATION ONLY  
CREATE UNIQUE INDEX index_name ON nickname  
(column_name DESC) SPECIFICATION ONLY
```

Bei einer Indexspezifikation ist *column_name* der Name, mit dessen Hilfe der Server mit zusammengeschlossenen Datenbanken auf eine Spalte einer Datenquellentabelle verweist.

Zugehörige Konzepte:

- „Indexspezifikationen in einem System zusammengeschlossener Datenbanken“ auf Seite 83

Zugehörige Tasks:

- „Erstellen von Indexspezifikationen für Tabellen, die neue Indizes erhalten“ auf Seite 86
- „Erstellen von Indexspezifikationen für Sichten“ auf Seite 88
- „Erstellen von Indexspezifikationen für Informix-Synonyme“ auf Seite 89

Zugehörige Referenzen:

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

Erstellen von Indexspezifikationen für Tabellen, die neue Indizes erhalten

Es gibt mehrere Situationen, in denen eine Tabelle einen neuen Index erhält:

- Sie erstellen einen Kurznamen für eine Tabelle, die keinen Index hat, zu einem späteren Zeitpunkt jedoch einen Index erhält.
- Sie erstellen einen Kurznamen für eine Tabelle, die einen Index hat, zu einem späteren Zeitpunkt jedoch einen anderen Index erhält.

In solchen Situationen sollten Sie eine Indexspezifikation für die Tabelle erstellen, sodass der SQL-Compiler diese Informationen bei der Verarbeitung von Abfragen, die auf diese Tabelle verweisen, verwenden kann.

Voraussetzungen:

Die Berechtigungs-ID der Anweisung muss zumindest über eines der folgenden Zugriffsrechte verfügen:

- Berechtigung SYSADM oder DBADM
- Entweder das Zugriffsrecht CONTROL oder das Zugriffsrecht INDEX für das Objekt. Außerdem muss die ID entweder über die Berechtigung IMPLICIT_SCHEMA für die Datenbank verfügen, wenn der implizite oder der explizite Schemaname des Indexes nicht vorhanden ist, oder über das Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname des Indexes auf ein bestehendes Schema bezieht.

Einschränkungen:

Bei der Erstellung eines Indexes für einen Kurznamen gibt es einige Einschränkungen.

- Wenn die Bindeoption DYNAMICRULES BIND zutrifft, kann die Anweisung nicht dynamisch vorbereitet werden. Außerdem können Sie die Parameter INCLUDE, CLUSTER, PCTFREE, MINPCTUSED, DISALLOW REVERSE SCANS und ALLOW REVERSE SCANS in der Anweisung CREATE INDEX nicht verwenden.
- UNIQUE sollte nur dann angegeben werden, wenn die Daten für den Indexschlüssel für jede Zeile der Datenquellentabelle eindeutige Werte enthalten. Die Eindeutigkeit wird nicht überprüft.

- Die Summe der gespeicherten Längen der angegebenen Spalten darf nicht größer als 1024 sein.
- Keine LOB- oder DATALINK-Spalte bzw. keine Spalte eines einzigartigen Datentyps auf der Basis von LOB oder DATALINK kann für einen Index verwendet werden. Diese Einschränkung wird auch dann durchgesetzt, wenn das Längensattribut der Spalte klein genug ist, damit die Begrenzung von 1024 Byte nicht überschritten wird.

Beispiel: Eine Tabelle hat keinen Index, erhält aber später einen Index:

Angenommen, Sie erstellen den Kurznamen *employee* für eine Datenquellentabelle mit dem Namen CURRENT_EMP, die keine Indizes hat. Einige Zeit nach der Erstellung dieses Kurznamens wird für CURRENT_EMP unter Verwendung der Spalten WORKDEPT und JOB für den Indexschlüssel ein Index definiert. Die Syntax zur Erstellung einer Indexspezifikation, die diesen Index beschreibt, lautet wie folgt:

```
CREATE UNIQUE INDEX job_by_dept ON employee
(WORKDEPT, JOB) SPECIFICATION ONLY
```

Dabei gilt: *job_by_dept* ist der Indexname.

Beispiel: Eine Tabelle erhält einen neuen Index:

Angenommen, Sie erstellen den Kurznamen *jp_sales* für eine Tabelle mit dem Namen JAPAN_SALES. Der Tabelle wird später zusätzlich zu den Indizes, die sie zum Zeitpunkt der Erstellung des Kurznamens hatte, ein neuer Index hinzugefügt. Der neue Index verwendet die Spalte MARKUP für den Indexschlüssel.

Die Syntax zur Erstellung einer Indexspezifikation, die diesen Index beschreibt, lautet wie folgt:

```
CREATE UNIQUE INDEX jp_markup ON jp_sales (MARKUP) SPECIFICATION ONLY
```

Dabei gilt: *jp_markup* ist der Indexname.

Zugehörige Konzepte:

- „Indexspezifikationen in einem System zusammengeschlossener Datenbanken“ auf Seite 83

Zugehörige Tasks:

- „Erstellen von Indexspezifikationen für Datenquellenobjekte“ auf Seite 84
- „Erstellen von Indexspezifikationen für Sichten“ auf Seite 88
- „Erstellen von Indexspezifikationen für Informix-Synonyme“ auf Seite 89

Zugehörige Referenzen:

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

Erstellen von Indexspezifikationen für Sichten

Der Server mit zusammengeschlossenen Datenbanken bemerkt beim Erstellen eines Kurznamens für eine Sicht nicht die zugrunde liegende Tabelle (und deren Indizes), aus der die Sicht generiert wurde. Erstellen Sie eine Indexspezifikation für die Sicht, sodass der SQL-Compiler diese Informationen bei der Verarbeitung von Abfragen, die auf diese Sicht verweisen, verwenden kann.

Voraussetzungen:

Die Berechtigungs-ID der Anweisung muss zumindest über eines der folgenden Zugriffsrechte verfügen:

- Berechtigung SYSADM oder DBADM
- Entweder das Zugriffsrecht CONTROL oder das Zugriffsrecht INDEX für das Objekt. Außerdem muss die ID entweder über die Berechtigung IMPLICIT_SCHEMA für die Datenbank verfügen, wenn der implizite oder der explizite Schemaname des Indexes nicht vorhanden ist, oder über das Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname des Indexes auf ein bestehendes Schema bezieht.

Einschränkungen:

Bei der Erstellung eines Indexes für einen Kurznamen gibt es einige Einschränkungen.

- Wenn die Bindeoption DYNAMICRULES BIND zutrifft, kann die Anweisung nicht dynamisch vorbereitet werden. Außerdem können Sie die Parameter INCLUDE, CLUSTER, PCTFREE, MINPCTUSED, DISALLOW REVERSE SCANS und ALLOW REVERSE SCANS in der Anweisung CREATE INDEX nicht verwenden.
- UNIQUE sollte nur dann angegeben werden, wenn die Daten für den Indexschlüssel für jede Zeile der Datenquellentabelle eindeutige Werte enthalten. Die Eindeutigkeit wird nicht überprüft.
- Die Summe der gespeicherten Längen der angegebenen Spalten darf nicht größer als 1024 sein.
- Keine LOB- oder DATALINK-Spalte bzw. keine Spalte eines einzigartigen Datentyps auf der Basis von LOB oder DATALINK kann für einen Index verwendet werden. Diese Einschränkung wird auch dann durchgesetzt, wenn das Längensattribut der Spalte klein genug ist, damit die Begrenzung von 1024 Byte nicht überschritten wird.

Kurzname wird für eine Sicht erstellt:

Angenommen, Sie erstellen den Kurznamen *jp_sales2003* für eine Sicht mit dem Namen JAPAN_SALES2003. Die zugrunde liegende Tabelle für diese Sicht ist die Tabelle JAPAN_SALES, die mehrere Indizes enthält: REGION, AMOUNT, SALES_REP. Die Anweisung CREATE INDEX, die Sie erstellen, verweist auf den Kurznamen für die Sicht und enthält Informationen zum Index der zugrunde liegenden Tabelle für die Sicht.

Wenn Sie eine Indexspezifikation für eine Sicht erstellen, müssen Sie sich vergewissern, dass die Spalte oder die Spalten, auf der/denen der Tabellenindex basiert, Teil der Sicht ist/sind. Wenn Sie für alle Indizes der zugrunde liegenden Tabelle Indexspezifikationen erstellen möchten, muss jede Indexspezifikation separat erstellt werden. Die Syntax zur Erstellung einer Indexspezifikation, die den Index REGION beschreibt, lautet beispielsweise wie folgt:

```
CREATE UNIQUE INDEX jp_2003_region ON jp_sales2003
(REGION) SPECIFICATION ONLY
```

Dabei gilt: *jp_2003_region* ist der Indexname und *jp_sales2003* ist der Kurzname für die Sicht JAPAN_SALES2003.

Zugehörige Konzepte:

- „Indexspezifikationen in einem System zusammenschlossener Datenbanken“ auf Seite 83

Zugehörige Tasks:

- „Erstellen von Indexspezifikationen für Datenquellenobjekte“ auf Seite 84
- „Erstellen von Indexspezifikationen für Tabellen, die neue Indizes erhalten“ auf Seite 86
- „Erstellen von Indexspezifikationen für Informix-Synonyme“ auf Seite 89

Zugehörige Referenzen:

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

Erstellen von Indexspezifikationen für Informix-Synonyme

In Informix können Sie für eine Tabelle oder Sicht ein Synonym erstellen. Der DB2-Server mit zusammenschlossenen Datenbanken ermöglicht zwar die Erstellung von Kurznamen für Informix-Synonyme, welche Aktion der Server mit zusammenschlossenen Datenbanken jedoch ausführt, hängt davon ab, ob das Synonym auf einer Tabelle oder auf einer Sicht basiert:

- Angenommen, für ein Synonym wird ein Kurzname erstellt, und das Synonym basiert auf einer Informix-Tabelle. Wenn der Server mit zusammenschlossenen Datenbanken ermittelt, dass die Tabelle, auf die sich das Synonym bezieht, einen Index hat, wird für das Synonym eine Indexspezifikation erstellt. Wenn die Tabelle, auf die sich das Synonym bezieht, keinen Index hat, wird für das Synonym keine Indexspezifikation erstellt. Sie können jedoch unter Verwendung der Anweisung CREATE INDEX manuell eine Indexspezifikation erstellen.
- Angenommen, für ein Synonym wird ein Kurzname erstellt, und das Synonym basiert auf einer Informix-Sicht. Der Server mit zusammenschlossenen Datenbanken kann nicht ermitteln, auf welcher/welchen zugrunde liegende(n) Tabelle(n) die Sicht basiert. Daher wird für das Synonym keine Indexspezifikation erstellt. Sie können jedoch unter Verwendung der Anweisung CREATE INDEX manuell eine Indexspezifikation erstellen.

Voraussetzungen:

Die Berechtigungs-ID der Anweisung muss zumindest über eines der folgenden Zugriffsrechte verfügen:

- Berechtigung SYSADM oder DBADM
- Entweder das Zugriffsrecht CONTROL oder das Zugriffsrecht INDEX für das Objekt. Außerdem muss die ID entweder über die Berechtigung IMPLICIT_SCHEMA für die Datenbank verfügen, wenn der implizite oder der explizite Schemaname des Indexes nicht vorhanden ist, oder über das Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname des Indexes auf ein bestehendes Schema bezieht.

Einschränkungen:

Bei der Erstellung eines Indexes für einen Kurznamen gibt es einige Einschränkungen.

- Wenn die Bindeoption DYNAMICRULES BIND zutrifft, kann die Anweisung nicht dynamisch vorbereitet werden. Außerdem können Sie die Parameter INCLUDE, CLUSTER, PCTFREE, MINPCTUSED, DISALLOW REVERSE SCANS und ALLOW REVERSE SCANS in der Anweisung CREATE INDEX nicht verwenden.
- UNIQUE sollte nur dann angegeben werden, wenn die Daten für den Indexschlüssel für jede Zeile der Datenquellentabelle eindeutige Werte enthalten. Die Eindeutigkeit wird nicht überprüft.
- Die Summe der gespeicherten Längen der angegebenen Spalten darf nicht größer als 1024 sein.
- Keine LOB- oder DATALINK-Spalte bzw. keine Spalte eines einzigartigen Datentyps auf der Basis von LOB oder DATALINK kann für einen Index verwendet werden. Diese Einschränkung wird auch dann durchgesetzt, wenn das Längensattribut der Spalte klein genug ist, damit die Begrenzung von 1024 Byte nicht überschritten wird.

Beispiel: Kurzname wird für ein Informix-Synonym erstellt, das auf einer Tabelle basiert:

Wenn das Synonym auf einer Informix-Tabelle basiert, die keinen Index enthält, können Sie für das Synonym eine Indexspezifikation erstellen, mit der dem Optimierungsprogramm mitgeteilt wird, welche Spalte(n) durchsucht werden muss/müssen, um Daten rasch zu finden. Die Anweisung, die Sie erstellen, gibt den Kurznamen für das Synonym an, und Sie stellen Informationen zu der Spalte/den Spalten in der Tabelle bereit, auf der/denen das Synonym basiert. Angenommen, Sie erstellen den Kurznamen *contracts* für ein Synonym mit dem Namen SALES_CONTRACTS, und die Tabelle, auf der dieses Synonym basiert, ist eine Tabelle mit mehreren Indizes: REGION, AMOUNT, SALES_REP. Die Anweisung CREATE INDEX, die Sie erstellen, verweist auf den Kurznamen für die Sicht und enthält Informationen zum Index der zugrunde liegenden Tabelle für die Sicht.

Beispiel: Kurzname wird für ein Informix-Synonym erstellt, das auf einer Sicht basiert:

Angenommen, Sie erstellen den Kurznamen *jp_sales2003* für eine Sicht mit dem Namen JAPAN_SALES2003. Die zugrunde liegende Tabelle für diese Sicht ist die Tabelle JAPAN_SALES, die mehrere Indizes enthält: REGION, AMOUNT, SALES_REP. Die Anweisung CREATE INDEX, die Sie erstellen, verweist auf den Kurznamen für die Sicht und enthält Informationen zum Index der zugrunde liegenden Tabelle für die Sicht.

Wenn Sie eine Indexspezifikation für eine Sicht erstellen, müssen Sie sich vergewissern, dass die Spalte oder die Spalten, auf der/denen der Tabellenindex basiert, Teil der Sicht ist/sind. Wenn Sie für alle Indizes der zugrunde liegenden Tabelle Indexspezifikationen erstellen möchten, muss jede Indexspezifikation separat erstellt werden.

Die Syntax zur Erstellung einer Indexspezifikation, die den Index REGION beschreibt, lautet wie folgt:

```
CREATE UNIQUE INDEX jp_2003_region ON jp_sales2003 (REGION) SPECIFICATION ONLY
```

Dabei gilt: *jp_2003_region* ist der Indexname und *jp_sales2003* ist der Kurzname für die Sicht JAPAN_SALES2003.

Zugehörige Konzepte:

- „Indexspezifikationen in einem System zusammengeschlossener Datenbanken“ auf Seite 83

Zugehörige Tasks:

- „Erstellen von Indexspezifikationen für Datenquellenobjekte“ auf Seite 84
- „Erstellen von Indexspezifikationen für Tabellen, die neue Indizes erhalten“ auf Seite 86
- „Erstellen von Indexspezifikationen für Sichten“ auf Seite 88

Zugehörige Referenzen:

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

Kapitel 6. Transparente DDL

In einem System zusammengeschlossener Datenbanken können Sie über den DB2-Server mit zusammengeschlossenen Datenbanken Linux, UNIX und Windows in den fernen Datenquellen Tabellen erstellen. Tabellen, die über den Server mit zusammengeschlossenen Datenbanken erstellt wurden, können über diesen Server auch geändert und gelöscht werden. Die Möglichkeit, über den DB2-Server mit zusammengeschlossenen Datenbanken ferne Tabellen zu erstellen, wird als *transparente DDL* bezeichnet.

In diesem Kapitel erhalten Sie Informationen zu folgenden Themen:

- „Was ist transparente DDL?“
- „Ferne LOB-Spalten und transparente DDL“ auf Seite 95
- „Erstellen neuer ferner Tabellen unter Verwendung von transparenter DDL“ auf Seite 95
- „Ändern von fernen Tabellen mit transparenter DDL“ auf Seite 99
- „Löschen von fernen Tabellen mit transparenter DDL“ auf Seite 101

Was ist transparente DDL?

| *Transparente DDL* bietet die Möglichkeit, ferne Tabellen ohne Verwendung von
| Durchgriffssitzungen mit DB2[®] Information Integrator zu erstellen und zu modifi-
| zieren.

Mit transparenter DDL werden die SQL-Anweisungen CREATE TABLE, ALTER TABLE und DROP TABLE verwendet.

| Eine Anweisung CREATE TABLE für transparente DDL erstellt in der Datenquelle
| eine ferne Tabelle und auf dem Server mit zusammengeschlossenen Datenbanken
| einen Kurznamen für diese Tabelle. Diese Anweisung ordnet die von Ihnen ange-
| gebenen DB2-Datentypen den fernen Datentypen unter Verwendung der standard-
| mäßigen regressiven Typzuordnungen zu. Für die meisten Datenquellen sind die
| Standardtypzuordnungen in den Wrappern enthalten. Die Standardtypzu-
| ordnungen für Datenquellen der DB2-Produktfamilie sind im DRDA[®]-Wrapper
| enthalten. Die Standardtypzuordnungen für Informix[®] sind im Informix-Wrapper
| enthalten, usw.

| Die Verwendung von transparenter DDL hat den Vorteil, dass DB2-Datenbank-
| administratoren für die Erstellung von sowohl lokalen als auch fernen Tabellen
| Prozeduren verwenden können, mit denen sie vertraut sind. "Transparente DDL"
| zentralisiert die Tabellenverwaltung und erleichtert das Erteilen von Berechtigun-
| gen.

| Transparente DDL wird für folgende Datenquellen unterstützt:

- DB2 für z/OS[™] und OS/390[®]
- DB2 für iSeries[™]
- DB2 für Linux, UNIX[®] und Windows[®]
- DB2 Server für VM und VSE
- Informix

- Microsoft® SQL Server
- ODBC
- Oracle
- Sybase
- Teradata

Der Datenbankadministrator kann entweder die DB2-Steuerzentrale oder DDL-Anweisungen im DB2-Befehlszeilenprozessor (CLP) für die Erstellung der Tabellen verwenden. Durch Verwendung von transparenter DDL muss er sich nicht die für jede Datenquelle jeweils erforderliche DDL-Syntax aneignen.

Bevor Sie mit DB2 Information Integrator für eine Datenquelle ferne Tabellen erstellen können, müssen Sie den Zugriff auf diese Datenquelle konfigurieren:

- Der Wrapper für diese Datenquelle muss im globalen Katalog registriert sein.
- Die Serverdefinition muss für den Server erstellt werden, auf dem die ferne Tabelle gespeichert wird.
- Die Benutzerzuordnungen zwischen DB2 Information Integrator und dem Datenquellenserver müssen erstellt werden.

Verwenden Sie für die Erstellung von fernen Tabellen den Assistenten für ferne Tabellen in der DB2-Steuerzentrale.

Die Zugriffsrechte der Berechtigungs-ID der transparenten DDL-Anweisungen müssen mindestens eines der folgenden Zugriffsrechte mit einschließen:

- Berechtigung SYSADM oder DBADM
- Berechtigung CREATETAB für die Datenbank und Zugriffsrecht USE für den Tabellenbereich sowie eine der folgenden Berechtigungen:
 - Berechtigung IMPLICIT_SCHEMA für die Datenbank, wenn der implizite oder der explizite Schemaname der Tabelle nicht vorhanden ist
 - Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname auf ein bestehendes Schema bezieht

Um transparente DDL-Anweisungen abzusetzen, muss Ihre Berechtigungs-ID die erforderlichen Zugriffsrechte für den Kurznamen haben (damit die zusammengeslossene Datenbank die Anfrage akzeptiert) und die vergleichbaren Zugriffsrechte für den fernen Datenquellenserver (damit die Datenquelle die Anfrage akzeptiert).

Transparente DDL - Einschränkungen

Für transparente DDL gelten einige Einschränkungen:

- Sie können Tabellen, die nativ in der fernen Datenquelle erstellt wurden, nicht ändern oder löschen.
- Gespeicherte Abfragetabellen können nicht in fernen Datenquellen erstellt werden.
- In der Tabellendefinition können grundlegende Spalteninformationen angegeben werden. Tabellen- oder Spaltenoptionen können darin jedoch nicht angegeben werden. Die LOB-Optionen (LOGGED und COMPACT) beispielsweise werden nicht unterstützt.
- Sie können keinen Kommentar für eine Spalte angeben.
- Sie können keine Spalteninhalte generieren.

- Sie können einen Primärschlüssel angeben. Sie können jedoch keinen Fremdschlüssel und keinen eindeutigen Schlüssel angeben und auch keine Prüfungen auf Integritätsbedingung durchführen. Die Spalten, die für einen Primärschlüssel verwendet werden, dürfen keine Nullwerte zulassen und dürfen keine Spalten sein, die LOBs enthalten.
- Sie können die Parameter bereits vorhandener Spalten, wie z. B. den Datentyp oder die Länge, nicht ändern.
- Die Klausel DEFAULT in den Anweisungen CREATE TABLE und ALTER TABLE wird nicht unterstützt.

Zugehörige Tasks:

- „Erstellen neuer ferner Tabellen unter Verwendung von transparenter DDL“ auf Seite 95
- „Ändern von fernen Tabellen mit transparenter DDL“ auf Seite 99
- „Löschen von fernen Tabellen mit transparenter DDL“ auf Seite 101

Ferne LOB-Spalten und transparente DDL

Einige Datenquellen, z. B. Oracle und Informix[®], speichern die Länge von LOB-Spalten nicht in ihren Systemkatalogen. Wenn Sie einen Kurznamen für eine Tabelle erstellen, werden Informationen aus dem Systemkatalog abgerufen, einschließlich der Spaltenlänge. Da für die LOB-Spalten keine Längenangabe vorhanden ist, nimmt die zusammengeschlossene Datenbank an, dass die Länge der maximalen Länge einer LOB-Spalte in DB2[®] für Linux, UNIX[®] und Windows[®] entspricht. Die zusammengeschlossene Datenbank speichert die in DB2 für Linux, UNIX und Windows maximal mögliche Länge als Länge der Kurznamenspalte im Katalog der zusammengeschlossenen Datenbank.

Wenn Sie jedoch eine ferne Tabelle mit transparenter DDL erstellen, müssen Sie die Länge der LOB-Spalte angeben. Wenn der Server mit zusammengeschlossenen Datenbanken einen Kurznamen für die ferne Tabelle erstellt, speichert er die von Ihnen angegebene Länge als Länge der Kurznamenspalte im Katalog der zusammengeschlossenen Datenbank.

Zugehörige Konzepte:

- „Was ist transparente DDL?“ auf Seite 93
- „LOB-Unterstützung in einem System zusammengeschlossener Datenbanken“ auf Seite 237

Zugehörige Tasks:

- „Erstellen neuer ferner Tabellen unter Verwendung von transparenter DDL“ auf Seite 95

Erstellen neuer ferner Tabellen unter Verwendung von transparenter DDL

Sie können zum Erstellen einer fernen Tabelle unter Verwendung von transparenter DDL entweder den Assistenten der DB2-Steuerzentrale oder die Anweisung CREATE TABLE verwenden.

Verwenden Sie den Assistenten für ferne Tabellen in der DB2-Steuerzentrale, um die Angabe eines nicht unterstützten Parameters oder einer nicht unterstützten

Option zu vermeiden. Mit dem Assistenten können Sie Spalten angeben, indem Sie aus einer Liste vordefinierter Spalten auswählen oder Attribute für eine neue Spalte angeben.

Voraussetzungen:

Bevor Sie eine ferne Tabelle erstellen, müssen Sie den Server mit zusammengesetzten Datenbanken für den Zugriff auf diese Datenquelle konfigurieren. Diese Konfiguration umfasst Folgendes:

- Erstellen des Wrappers für den Datenquellentyp
- Bereitstellen der Serverdefinition für den Server, auf dem die ferne Tabelle gespeichert werden wird
- Erstellen der Benutzerzuordnungen zwischen DB2 Information Integrator und dem Datenquellenserver

Um transparente DDL-Anweisungen abzusetzen, muss Ihre Berechtigungs-ID die erforderlichen Zugriffsrechte für den Kurznamen haben (damit die zusammengesetzte Datenbank die Anfrage akzeptiert) und die vergleichbaren Zugriffsrechte für den fernen Datenquellenserver (damit die Datenquelle die Anfrage akzeptiert).

Die Zugriffsrechte der Berechtigungs-ID der transparenten DDL-Anweisungen müssen mindestens eine der folgenden Berechtigungen umfassen:

- Berechtigung SYSADM oder DBADM
- Berechtigung CREATETAB für die Datenbank und Zugriffsrecht USE für den Tabellenbereich sowie eine der folgenden Berechtigungen:
 - Berechtigung IMPLICIT_SCHEMA für die Datenbank, wenn der implizite oder der explizite Schemaname der Tabelle nicht vorhanden ist
 - Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname auf ein bestehendes Schema bezieht

Einschränkungen:

Gespeicherte Abfragetabellen können nicht in fernen Datenquellen erstellt werden.

In der Tabellendefinition können grundlegende Spalteninformationen angegeben werden. Tabellen- oder Spaltenoptionen können darin jedoch nicht angegeben werden. Die LOB-Optionen (LOGGED und COMPACT) beispielsweise werden nicht unterstützt.

Sie können keinen Kommentar für eine Spalte angeben.

Sie können keine Spalteninhalte generieren.

Sie können einen Primärschlüssel angeben. Sie können jedoch keinen Fremdschlüssel angeben und keine Prüfungen auf Integritätsbedingung durchführen. Die Spalten, die für einen Primärschlüssel verwendet werden, dürfen keine Nullwerte zulassen und dürfen keine Spalten sein, die LOBs enthalten.

Die Klausel DEFAULT wird in der Anweisung CREATE TABLE nicht unterstützt.

Vorgehensweise:

Sie können eine ferne Tabelle unter Verwendung transparenter DDL von der DB2-Steuerzentrale oder von der Eingabeaufforderung der Befehlszeile aus erstellen.

Verwenden Sie den Assistenten zum Erstellen ferner Tabellen, um von der DB2-Steuerzentrale aus eine ferne Tabelle zu erstellen:

1. Erweitern Sie den Ordner **Objekte zusammengeschlossener Datenbanken**.
2. Erweitern Sie den Wrapper und die Serverdefinitionsobjekte für die Datenquelle, für die Sie eine ferne Tabelle erstellen möchten.
3. Klicken Sie mit der rechten Maustaste auf den Ordner **Ferne Tabellen**, und klicken Sie auf **Erstellen**. Der Assistent **Ferne Tabelle erstellen** wird gestartet.
4. Führen Sie die Schritte im Assistenten aus.

Geben Sie die Anweisung CREATE TABLE ein, um diese Task von der Eingabeaufforderung der Befehlszeile aus auszuführen; dabei müssen die entsprechenden Parameter definiert sein.

Die ferne Datenquelle muss die Spaltendatentypen und die Primärschlüsseloption in der Anweisung CREATE TABLE unterstützen. Beispiel: Angenommen, die ferne Datenquelle unterstützt keine Primärschlüssel. Je nachdem wie die Datenquelle auf Anfragen antwortet, die sie nicht unterstützt, wird möglicherweise ein Fehler zurückgegeben oder die Anfrage wird ignoriert.

Der ferne Server muss in der Klausel OPTIONS angegeben sein. Die Klausel OPTIONS kann verwendet werden, um die fernen Namen oder das ferne Schema der zu erstellenden Tabelle außer Kraft zu setzen.

Angenommen, Sie möchten auf einem Oracle-Server die Tabelle EMPLOYEE erstellen. Verwenden Sie in der Anweisung CREATE TABLE die DB2-Datentypen, wenn Sie die einzelnen Spalten angeben. Bei der Verwendung des Befehlszeilenprozessors (CLP) zum Erstellen der Tabelle lautet die Syntax wie folgt:

```
CREATE TABLE EMPLOYEE
( EMP_NO      CHAR(6) NOT NULL,
  FIRSTNAME   VARCHAR(12) NOT NULL,
  MIDINT      CHAR(1) NOT NULL,
  LASTNAME    VARCHAR(15) NOT NULL,
  HIREDATE    DATE,
  JOB         CHAR(8),
  SALARY      DECIMAL(9,2),
  PRIMARY KEY (EMP_NO) )
OPTIONS (REMOTE_SERVER 'ORASERVER',
        REMOTE_SCHEMA 'J15USER1', REMOTE_TABNAME 'EMPLOY' )
EMPLOYEE
```

Der lokale Tabellename. Dieser Name wird auch für den Kurznamen verwendet, der der Tabelle zugeordnet ist.

REMOTE_SERVER 'ORASERVER'

Der Name, den Sie für den Server in der Anweisung CREATE SERVER angegeben haben. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

REMOTE_SCHEMA 'J15USER1'

Der Name des fernen Schemas. Dieser Parameter ist zwar optional, es wird jedoch empfohlen, dass Sie einen Schemanamen angeben. Wenn dieser Parameter nicht angegeben wird, wird der lokale Parameter AUTHID in Großschreibung als ferner Schemaname verwendet. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

REMOTE_TABNAME 'EMPLOY'

Der Name der fernen Tabelle. Die Angabe dieses Parameters ist optional. Wenn dieser Parameter nicht angegeben wird, wird der lokale Tabellenname als ferner Tabellenname verwendet. Dieser Wert muss ein gültiger Name in der fernen Datenquelle sein und darf kein vorhandener Tabellenname sein. Bei diesem Wert muss die Groß-/Kleinschreibung beachtet werden.

Wenn eine ferne Tabelle mit DB2 Information Integrator unter Verwendung von transparenter DDL erstellt wird, werden auch einige andere Aktionen ausgeführt:

- Für die ferne Tabelle wird automatisch ein Kurzname erstellt. Der Kurzname stimmt mit dem Namen der lokalen Tabelle überein. Die ferne Tabelle hat denselben Namen wie die lokale Tabelle, es sei denn, Sie geben unter Verwendung der Option REMOTE_TABNAME einen anderen Namen an. Der Kurzname für die ferne Tabelle kann wie jeder andere Kurzname verwendet werden. Außerdem können Sie die ferne Tabelle ändern (ALTER) und löschen (DROP). (Dies können Sie mit einem Kurznamen nicht machen, der mit CREATE NICKNAME erstellt wurde.)
- In der Katalogsicht SYSCAT.TABOPTIONS wird eine Zeile mit dem Optionsnamen TRANSPARENT und dem Wert 'Y' hinzugefügt.

Im obigen Beispiel verwendet DB2 Information Integrator regressive Datentypzuordnungen, um die DB2-Datentypen zu Oracle-Datentypen zuzuordnen. Auf dem fernen Oracle-Server wird die Tabelle EMPLOY mit Oracle-Datentypen erstellt. In der folgenden Tabelle finden Sie die Zuordnungen der DB2-Datentypen zu Oracle-Datentypen für die im Beispiel angegebenen Spalten.

Tabelle 6. Beispiel für regressive Datentypzuordnungen von DB2 Information Integrator zu Oracle

Spalte	In der Anweisung CREATE TABLE angegebener DB2-Datentyp	In der fernen Tabelle verwendeter Oracle-Datentyp
EMP_NO	CHAR(6) NOT NULL	CHAR(6) NOT NULL
FIRST_NAME	VARCHAR(12) NOT NULL	VARCHAR2(12) NOT NULL
MID_INT	CHAR(1) NOT NULL	CHAR(1) NOT NULL
LAST_NAME	VARCHAR(15) NOT NULL	VARCHAR2(15) NOT NULL
HIRE_DATE	DATE	DATE
JOB	CHAR(8)	CHAR(8)
SALARY	DECIMAL(9,2)	NUMBER(9,2)

Die Option SQL_SUFFIX ist am Ende der Anweisung CREATE TABLE zulässig. Diese Option wird hauptsächlich zur Angabe der Klausel IN TABLESPACE beim Erstellen ferner Tabellen in Datenquellen der DB2-Produktfamilie verwendet.

Zugehörige Konzepte:

- „Was ist transparente DDL?“ auf Seite 93

Zugehörige Tasks:

- „Ändern von fernen Tabellen mit transparenter DDL“ auf Seite 99
- „Löschen von fernen Tabellen mit transparenter DDL“ auf Seite 101

Zugehörige Referenzen:

- „CREATE TABLE statement“ in *SQL Reference, Volume 2*
- Kapitel 28, „Standardmäßige regressive Typzuordnungen“, auf Seite 315

Ändern von fernen Tabellen mit transparenter DDL

Sie können ferne Datenquellentabellen ändern, die mit DB2 Information Integrator unter Verwendung von transparenter DDL erstellt wurden. Sie können keine Tabellen ändern, die direkt in der fernen Datenquelle erstellt wurden.

Verwenden Sie die Anweisung ALTER TABLE, um Tabellen zu ändern, die mit DB2 Information Integrator unter Verwendung von transparenter DDL erstellt wurden. Mit der Anweisung ALTER TABLE können Sie folgende Aktionen ausführen:

- Neue Spalten hinzufügen
- Den Primärschlüssel der Tabelle hinzufügen

Voraussetzungen:

Die Zugriffsrechte der Berechtigungs-ID der transparenten DDL-Anweisungen müssen mindestens eine der folgenden Berechtigungen umfassen:

- Berechtigung SYSADM oder DBADM
- Berechtigung CREATETAB für die Datenbank und Zugriffsrecht USE für den Tabellenbereich sowie eine der folgenden Berechtigungen:
 - Berechtigung IMPLICIT_SCHEMA für die Datenbank, wenn der implizite oder der explizite Schemaname der Tabelle nicht vorhanden ist
 - Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname auf ein bestehendes Schema bezieht

Um transparente DDL-Anweisungen abzusetzen, muss Ihre Berechtigungs-ID die erforderlichen Zugriffsrechte für den Kurznamen haben (damit die zusammengesetzte Datenbank die Anfrage akzeptiert) und die vergleichbaren Zugriffsrechte für den fernen Datenquellenserver (damit die Datenquelle die Anfrage akzeptiert).

Einschränkungen:

Sie können keine Tabellen ändern, die in der fernen Datenquelle erstellt wurden.

Ein vorhandener Primärschlüssel kann nicht in einer fernen Tabelle geändert oder gelöscht werden.

Das Ändern einer fernen Tabelle macht alle Pakete ungültig, die von dem Kurznamen abhängig sind, der der fernen Tabelle zugeordnet ist.

Die ferne Datenquelle muss die Änderungen in der Anweisung ALTER TABLE unterstützen. Beispiel: Angenommen, die ferne Datenquelle unterstützt keine Primärschlüssel. Je nachdem wie die Datenquelle auf Anfragen antwortet, die sie nicht unterstützt, wird möglicherweise ein Fehler zurückgegeben oder die Anfrage wird ignoriert.

Sie können keinen Kommentar für eine Spalte angeben.

Sie können keine Spalteninhalte generieren.

Sie können einen Primärschlüssel angeben. Sie können jedoch keinen Fremdschlüssel angeben und keine Prüfung auf Integritätsbedingungen durchführen. Die Spalten, die für einen Primärschlüssel verwendet werden, dürfen keine Nullwerte zulassen und dürfen keine Spalten sein, die LOBs enthalten.

Sie können die Parameter von bereits vorhandenen Spalten, wie z. B. den Datentyp oder die Länge, nicht ändern.

Die Klausel DEFAULT wird in der Anweisung ALTER TABLE nicht unterstützt.

Vorgehensweise:

Sie können zum Ändern einer fernen Tabelle unter Verwendung von transparenter DDL entweder die DB2-Steuerzentrale oder die Anweisung ALTER TABLE verwenden. Verwenden Sie die DB2-Steuerzentrale, um die Angabe eines nicht unterstützten Parameters oder einer nicht unterstützten Option zu vermeiden.

Verwenden Sie die Anweisung ALTER TABLE nicht, um Spaltenoption hinzuzufügen oder zu ändern. Verwenden Sie stattdessen die Anweisung ALTER NICKNAME.

Angenommen, Sie möchten einen Primärschlüssel für eine ferne Tabelle EMPLOYEE hinzufügen, die Sie mit transparenter DDL erstellt haben. Bei der Verwendung des Befehlszeilenprozessors (CLP) zum Ändern der Tabelle lautet die Syntax wie folgt:

```
ALTER TABLE EMPLOYEE
    ADD PRIMARY KEY (EMP_NO, WORK_DEPT)
```

Die Spalten, die für einen Primärschlüssel verwendet werden, dürfen keine Nullwerte (NOT NULL) zulassen und keine LOBs enthalten.

Angenommen, Sie möchten die Spalten ORDER_DATE und SHIP_DATE zu der fernen Tabelle SPALTEN hinzufügen, und diese Tabelle wurde mit transparenter DDL erstellt. Bei der Verwendung des Befehlszeilenprozessors (CLP) zum Erstellen der Tabelle lautet die Syntax wie folgt:

```
ALTER TABLE SPALTEN
    ADD COLUMN ORDER_DATE DATE
    ADD COLUMN SHIP_DATE DATE
```

Zugehörige Konzepte:

- „Was ist transparente DDL?“ auf Seite 93

Zugehörige Tasks:

- „Erstellen neuer ferner Tabellen unter Verwendung von transparenter DDL“ auf Seite 95
- „Löschen von fernen Tabellen mit transparenter DDL“ auf Seite 101

Zugehörige Referenzen:

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

Löschen von fernen Tabellen mit transparenter DDL

Sie können ferne Datenquellentabellen löschen, die mit DB2 Information Integrator unter Verwendung von transparenter DDL erstellt wurden. Sie können keine Tabellen löschen, die direkt in der fernen Datenquelle erstellt wurden.

Voraussetzungen:

Die Zugriffsrechte der Berechtigungs-ID der transparenten DDL-Anweisungen müssen mindestens eine der folgenden Berechtigungen umfassen:

- Berechtigung SYSADM oder DBADM
- Berechtigung CREATETAB für die Datenbank und Zugriffsrecht USE für den Tabellenbereich sowie eine der folgenden Berechtigungen:
 - Berechtigung IMPLICIT_SCHEMA für die Datenbank, wenn der implizite oder der explizite Schemaname der Tabelle nicht vorhanden ist
 - Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname auf ein bestehendes Schema bezieht

| Um transparente DDL-Anweisungen abzusetzen, muss Ihre Berechtigungs-ID die
| erforderlichen Zugriffsrechte für den Kurznamen haben (damit die zusammenge-
| schlossene Datenbank die Anfrage akzeptiert) und die vergleichbaren Zugriffs-
| rechte für den fernen Datenquellenserver (damit die Datenquelle die Anfrage
| akzeptiert).

Einschränkungen:

Sie können keine Tabellen löschen, die in der fernen Datenquelle erstellt wurden.

Vorgehensweise:

| Sie können zum Löschen einer fernen Tabelle, die mit DB2 Information Integrator
| unter Verwendung von transparenter DDL erstellt wurde, entweder die DB2-
| Steuerzentrale oder die Anweisung DROP verwenden. Das Löschen eines Kurz-
| namens für eine ferne Tabelle, die mit transparenter DDL erstellt wurde, löscht nur
| den lokalen Kurznamen für diese Tabelle. Die Anweisung DROP NICKNAME
| löscht nicht die ferne Tabelle. Sie müssen die Anweisung DROP TABLE verwen-
| den, um die ferne Tabelle zu löschen.

Das Löschen einer fernen Tabelle löscht zuerst die Tabelle in der Datenquelle und danach den Kurznamen für die ferne Tabelle in der zusammengeschlossenen Datenbank. Durch das Löschen des Kurznamens wird die Gültigkeit aller Pakete aufgehoben, die auf diesem Kurznamen basieren.

Die Syntax zum Löschen der Tabelle SPALTEN lautet wie folgt:

```
DROP TABLE SPALTEN
```

| Dabei ist *SPALTEN* der lokale Name der fernen Tabelle.

Zugehörige Konzepte:

- „Was ist transparente DDL?“ auf Seite 93

Zugehörige Tasks:

- „Erstellen neuer ferner Tabellen unter Verwendung von transparenter DDL“ auf Seite 95
- „Ändern von fernen Tabellen mit transparenter DDL“ auf Seite 99

Zugehörige Referenzen:

- „DROP statement“ in *SQL Reference, Volume 2*

Kapitel 7. Transaktionsunterstützung in einem System zusammengeschlossener Datenbanken

Die Transaktionsunterstützung in einem System zusammengeschlossener Datenbanken baut auf der Transaktionsfunktionalität einer verteilten DB2-Datenbank auf. In diesem Kapitel wird vorausgesetzt, dass Sie über Kenntnisse in den DB2-Basiskonzepten der Verarbeitung verteilter Transaktionen verfügen, die in den folgenden DB2-Handbüchern beschrieben sind:

- *IBM DB2 UDB Systemverwaltung: Konzept (IBM Form SC12-3057)*
- *DB2 Application Development Guide: Programming Client Applications (IBM Form SC09-4826)*
- *DB2 Application Development Guide: Programming Server Applications (IBM Form SC09-4827)*

Dieses Kapitel enthält die folgenden Themen:

- „Transaktionsunterstützung in einem System zusammengeschlossener Datenbanken“
- „Was ist eine Aktualisierung in einem System zusammengeschlossener Datenbanken?“ auf Seite 105

Transaktionsunterstützung in einem System zusammengeschlossener Datenbanken

Die Transaktionsverarbeitung in einem System zusammengeschlossener Datenbanken baut auf der Transaktionsfunktionalität einer verteilten DB2[®]-Datenbank auf. Damit Sie sich einen Überblick über die Transaktionsverarbeitung in einem System zusammengeschlossener Datenbanken verschaffen können, müssen Sie mit den folgenden wesentlichen Konzepten der verteilten Transaktionsverarbeitung in DB2 vertraut sein:

- Arbeitseinheit (UOW)
- Ferne Arbeitseinheit (RUOW)
- Verteilte Arbeitseinheit (DUOW)
- Aktualisierung auf mehreren Systemen
- Transaktionsmanager (TM)
- Ressourcenmanager (RM)
- Verbindung des Typs 1
- Verbindung des Typs 2
- Einphasige Festschreibung
- Zweiphasige Festschreibung

Diese Konzepte funktionieren sowohl in DB2-Systemen zusammengeschlossener Datenbanken als auch in DB2-Systemen ohne zusammengeschlossene Datenbanken. Der Geltungsbereich der einzelnen Konzepte ändert sich jedoch in Systemen zusammengeschlossener Datenbanken.

Beispiel: Eine Arbeitseinheit wird implizit begonnen, wenn für beliebige Daten einer Datenbank ein Lese- oder Schreibvorgang ausgeführt wird. Für eine Arbeitseinheit in einem System zusammengeschlossener Datenbanken kann die Datenbank

| eine zusammengeschlossene Datenbank oder eine Datenquellendatenbank sein. Für
| eine verteilte Arbeitseinheit in einem System zusammengeschlossener Datenbanken
| ist der Zugriff auf eine zusammengeschlossene Datenbank und eine Datenquellen-
| datenbank möglich.

Eine Anwendung muss eine Arbeitseinheit durch Absetzen einer Anweisung COMMIT oder ROLLBACK beenden, unabhängig von der Anzahl Datenbanken, auf die zugegriffen wird. Die Anweisung COMMIT schreibt alle innerhalb einer Arbeitseinheit ausgeführten Änderungen fest und macht sie dadurch permanent. Die Anweisung ROLLBACK entfernt diese Änderungen aus einer Datenbank. Die durch eine Arbeitseinheit ausgeführten Änderungen werden nach einer erfolgreichen COMMIT-Operation in anderen Anwendungen sichtbar.

Empfehlung: Arbeitseinheiten sollten von Anwendungen immer explizit festgeschrieben oder rückgängig gemacht werden.

| Wenn nur eine Datenquelle in einer Arbeitseinheit aktualisiert wird, kann eine
| Anweisung COMMIT für diese Arbeitseinheit in einer einzelnen Operation gesen-
| det werden. Diese Operation wird als *einphasige Festschreibung* oder Aktualisierung
| auf einem einzelnen System bezeichnet. Ein *System* entspricht einer Serverdefi-
| nition in einem System zusammengeschlossener Datenbanken. Ein Server mit zu-
| sammengeschlossenen Datenbanken bildet das lokale System für Aktualisie-
| rungsoperationen in einem System zusammengeschlossener Datenbanken. Alle fer-
| nen Datenquellen gelten für Aktualisierungsoperationen in einem System zusam-
| mengeschlossener Datenbanken als ferne Systeme.

In einer einphasigen Festschreibungsoperation sind in derselben Arbeitseinheit keine weiteren Datenquellen erforderlich, um deren Fähigkeit zum Festschreiben von Daten zu prüfen.

Nachfolgend sind einige Beispiele für eine einphasige Festschreibung aufgeführt:

- Nicht verteilte Arbeitseinheit
- Eine verteilte Arbeitseinheit, in der in einer oder mehreren Datenquellen Lesevorgänge durchführt, aber nur in einer einzigen Datenquelle Daten aktualisiert werden

| In einer verteilten Arbeitseinheit, in der mehrere Datenbanken in mehreren Systeme
| aktualisiert werden, müssen die Daten konsistent sein. Das Protokoll für die
| Aktualisierung auf mehreren Systemen bzw. für die zweiphasige Festschreibung
| wird allgemein verwendet, um die Datenkonsistenz mehrerer Datenbanken inner-
| halb einer verteilten Arbeitseinheit zu gewährleisten. Systeme zusammengeschlos-
| sener Datenbanken bieten derzeit keine Unterstützung für das Protokoll für zwei-
| phasige Festschreibung.

Zugehörige Konzepte:

- „Arbeitseinheiten“ in *Systemverwaltung: Konzept*
- „DB2-Transaktionsmanager“ in *Systemverwaltung: Konzept*
- „Zweiphasige Festschreibung“ in *Systemverwaltung: Konzept*
- „Einrichten von Ressourcenmanagern“ in *Systemverwaltung: Konzept*
- „Multisite Update“ in *Application Development Guide: Programming Client Applications*
- „Was ist eine Aktualisierung in einem System zusammengeschlossener Datenbanken?“ auf Seite 105

Zugehörige Tasks:

- „Aktualisieren einer Datenbank von einem Host- oder iSeries-Client“ in *Systemverwaltung: Konzept*
- „Aktualisieren einer Einzeldatenbank in einer Transaktion“ in *Systemverwaltung: Konzept*

Was ist eine Aktualisierung in einem System zusammengeschlossener Datenbanken?

In einem System zusammengeschlossener Datenbanken ist eine Aktualisierung nicht nur eine Transaktion, die eine Anweisung des Typs INSERT, UPDATE oder DELETE enthält. Der Server mit zusammengeschlossenen Datenbanken betrachtet bestimmte Aktionen ebenfalls als Aktualisierungen.

Aktualisierungen können lokal oder fern ausgeführt werden.

Lokale Aktualisierungen sind Aktualisierungen von Objekten in der zusammengeschlossenen Datenbank. Tabellen und Sichten in der zusammengeschlossenen Datenbank sind lokale Objekte. Für diese Objekte erstellen Sie keine Kurznamen, sondern Sie verwenden in den Anweisungen den eigentlichen Objektnamen.

Ferne Aktualisierungen sind Aktualisierungen von Objekten in einer fernen Datenquelle. Zu den fernen Datenquellen gehören:

- Andere Datenbanken oder Exemplare von DB2[®] für Linux, UNIX[®] und Windows[®] auf dem Server mit zusammengeschlossenen Datenbanken
- Andere Datenbanken oder Exemplare von DB2 für Linux, UNIX und Windows auf einem anderen Server
- Andere Datenquellen als DB2 für Linux, UNIX und Windows, z. B. DB2 für iSeries[™], Informix[®], Oracle und Teradata

Es ist wichtig, dass Sie wissen, welche Transaktionen der Server mit zusammengeschlossenen Datenbanken als Aktualisierungen betrachtet und welche Aktualisierungstypen in einem System zusammengeschlossener Datenbanken zulässig sind. Es gibt vier Aktionstypen, die der Server mit zusammengeschlossenen Datenbanken als Aktualisierungstransaktionen betrachtet. In der folgenden Tabelle sind die Aktualisierungen aufgeführt, die Sie in einem System zusammengeschlossener Datenbanken ausführen können.

Tabelle 7. Aktualisierungstypen und das jeweilige System, in dem die Aktualisierungen ausgeführt werden

Aktionstyp	Lokales System	Fernes System	Erläuterung
Lokale Aktualisierung (DDL und DML)	Y	N	Eine Aktualisierung eines Objekts in der zusammengeschlossenen Datenbank.
Ferne Aktualisierung (Kurzname)	N	Y	Eine Aktualisierung eines fernen Datenquellenobjekts, für das Sie einen Kurznamen erstellt haben.
Dynamisches SQL in Durchgriffssitzungen	N	Y	Eine Aktualisierung eines fernen Datenquellenobjekts. Zum Aktualisieren von lokalen Objekten können Sie keine Durchgriffssitzungen verwenden.

Tabelle 7. Aktualisierungstypen und das jeweilige System, in dem die Aktualisierungen ausgeführt werden (Forts.)

Aktionstyp	Lokales System	Fernes System	Erläuterung
Transparente DDL	Y	Y	Ein Transaktionspaar, mit dem ferne Tabellen und deren zugehörige Kurznamen in einer zusammengeschlossenen Datenbank erstellt, geändert oder gelöscht werden können. Beispielsweise ein Transaktionspaar, mit dem eine ferne Tabelle in einer Datenquelle und ein Kurzname auf dem Server mit zusammengeschlossenen Datenbanken erstellt werden.

Was ist eine Aktualisierungstransaktion in einer Durchgriffssitzung?

Ein Server mit zusammengeschlossenen Datenbanken behandelt alle dynamischen SQL-Anweisungen, die über Durchgriffssitzungen gesendet werden, als Aktualisierungen. Dadurch wird die Datenintegrität gewährleistet. Wenn eine dynamische SQL-Anweisung, die über eine Durchgriffssitzung gesendet wird, erfolgreich ist, wird die Transaktion als Aktualisierung aufgezeichnet. Die SQL-Anweisung kann jeden beliebigen Typ haben, einschließlich SELECT.

Transaktionsunterstützung bei transparenter DDL

Die transparente DDL erstellt eine Tabelle in der fernen Datenquelle sowie einen Kurznamen für die ferne Tabelle in der lokalen zusammengeschlossenen Datenbank. Ein Server mit zusammengeschlossenen Datenbanken behandelt transparente DDL-Transaktionen als Aktualisierungen.

Vor und nach einer transparenten DDL-Transaktionen muss eine Anweisung COMMIT oder ROLLBACK abgesetzt werden.

Da die transparente DDL gleichzeitig lokale und ferne Objekte erstellt, muss jede abgesetzte transparente DDL-Anweisung jeweils die einzige Aktualisierung innerhalb einer einphasigen Festschreibungstransaktion sein. Geht der transparenten DDL-Transaktion eine andere Aktualisierung voraus, muss vor der transparenten DDL-Transaktion eine Anweisung COMMIT oder ROLLBACK ausgegeben werden. Ebenso muss nach der transparenten DDL-Transaktion eine Anweisung COMMIT oder ROLLBACK ausgegeben werden, bevor eine weitere Aktualisierung ausgeführt werden kann.

Datenquellen, die DDL-Anweisungen automatisch festschreiben

Einige Datenquellen schreiben die aktuelle Transaktion nach einer erfolgreichen DDL-Transaktion automatisch lokal fest. Wenn Sie mit einer transparenten DDL-Transaktion oder in einer Durchgriffssitzung eine ferne Tabelle erstellen, können diese Datenquellen nach der Erstellung der Tabelle keine ROLLBACK-Operation für die ferne Tabelle ausführen. Sie müssen die ferne Tabelle manuell löschen.

Die einzige zusammengeschlossene Datenquelle, die DDL-Anweisungen automatisch festschreibt, ist Oracle.

Benutzerdefinierte Funktionen, die im Pushdown-Modus zur Verarbeitung an die Datenquelle weitergeleitet werden

Ein Server mit zusammengeschlossenen Datenbanken behandelt benutzerdefinierte Funktionen, die im Pushdown-Modus zur Verarbeitung an die Datenquelle weitergeleitet werden, als Anweisungen mit Lesezugriff. Wenn eine ferne benutzerdefinierte Funktion eine Aktualisierung in einer Datenquelle ausführt, bemerkt der Server mit zusammengeschlossenen Datenbanken die Aktualisierung nicht. Da der Server mit zusammengeschlossenen Datenbanken diese benutzerdefinierten Funktionen nicht als Aktualisierungsanweisungen betrachtet, wird die Anweisung nicht rückgängig gemacht.

Wichtig: Wenn eine benutzerdefinierte Funktion, die im Pushdown-Modus zur Verarbeitung an eine Datenquelle weitergeleitet wird, eine Aktualisierung ausführt, kann die Datenintegrität nicht gewährleistet werden.

Zugehörige Konzepte:

- „Direktes Abfragen von Datenquellen im Durchgriffsmodus“ auf Seite 242
- „Transaktionsunterstützung in einem System zusammengeschlossener Datenbanken“ auf Seite 103

Kapitel 8. INSERT-, UPDATE- und DELETE-Operationen

In diesem Kapitel wird beschrieben, wie Sie auf Daten in den Datenquellen zugreifen und diese aktualisieren können.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Zugriffsrechte für die Anweisungen INSERT, UPDATE und DELETE“
- „Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammenschlossener Datenbanken“ auf Seite 110
- „Referenzielle Integrität in einem System zusammenschlossener Datenbanken“ auf Seite 111
- „Anweisungen INSERT, UPDATE und DELETE für große Objekte (LOBs)“ auf Seite 112
- „Gewährleisten der Anweisungsautarkie in einem System zusammenschlossener Datenbanken“ auf Seite 112
- „Arbeiten mit Kurznamen“ auf Seite 114
- „Auswählen von Daten in einem System zusammenschlossener Datenbanken“ auf Seite 126
- „Einfügen von Daten in Datenquellenobjekte“ auf Seite 130
- „Aktualisieren von Daten in Datenquellenobjekten“ auf Seite 131
- „Löschen von Daten aus Datenquellenobjekten“ auf Seite 132

Zugriffsrechte für die Anweisungen INSERT, UPDATE und DELETE

Die zum Absetzen der Anweisungen INSERT, UPDATE und DELETE für Kurznamen erforderlichen Zugriffsrechte ähneln den Zugriffsrechten, die zum Absetzen dieser Anweisungen für Tabellen erforderlich sind. Sie müssen über die erforderlichen Zugriffsrechte für die Datenquelle verfügen, um für das zugrunde liegende Objekt Operationen des Typs SELECT, INSERT, UPDATE oder DELETE ausführen zu können.

Sie können die Zugriffsrechte für SELECT, INSERT, UPDATE und DELETE für Kurznamen erteilen oder widerrufen.

Das Erteilen oder Widerrufen von Zugriffsrechten für einen Kurznamen führt nicht zum Erteilen oder Widerrufen von Zugriffsrechten für die Datenquelle. In der Datenquelle müssen die Zugriffsrechte für die Berechtigungs-ID REMOTE_AUTHID gewährt oder widerrufen werden, die in der Benutzerzuordnung auf dem Server mit zusammenschlossenen Datenbanken angegeben ist.

Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen die erforderlichen Zugriffsrechte für den Kurznamen umfassen, damit die zusammenschlossene Datenbank die Anforderung akzeptiert. Die Benutzer-ID in der Datenquelle, die (durch eine Benutzerzuordnung) der Berechtigungs-ID zugeordnet ist, muss über die erforderlichen Zugriffsrechte für das zugrunde liegende Tabellenobjekt verfügen, damit die Datenquelle die Anforderung akzeptiert.

Wird eine Abfrage an die zusammenschlossene Datenbank übergeben, werden die Zugriffsrechte für den Kurznamen in der Abfrage überprüft.

Die Berechtigungsanforderungen des Datenquellenobjekts, auf das der Kurzname verweist, finden nur während der Verarbeitung der Abfrage Anwendung. Wenn Sie nicht über das Zugriffsrecht SELECT für den Kurznamen verfügen, können Sie aus dem Objekt, auf das der Kurzname verweist, keine Auswahl treffen.

Ebenso bedeutet das Vorhandensein des Zugriffsrechts UPDATE für einen Kurznamen nicht, dass Sie automatisch zur Aktualisierung des durch den Kurznamen dargestellten Objekts berechtigt sind. Eine erfolgreiche Überprüfung der Zugriffsrechte auf dem Server mit zusammengeschlossenen Datenbanken bedeutet nicht, dass auch die Überprüfung der Zugriffsrechte in der fernen Datenquelle erfolgreich ist. Die Berechtigungs-ID des Servers mit zusammengeschlossenen Datenbanken wird mit Hilfe von Benutzerzuordnungen der Benutzer-ID der fernen Datenquelle zugeordnet. Die Einschränkungen der Zugriffsrechte werden in der Datenquelle erzwungen.

Zugehörige Tasks:

- „Ändern einer Benutzerzuordnung“ auf Seite 37
- „Mit Kurznamen verwendbare SQL-Anweisungen“ auf Seite 116

Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammengeschlossener Datenbanken

In Systemen zusammengeschlossener Datenbanken ist eine Aktualisierung nicht nur eine Transaktion, die eine Anweisung des Typs INSERT, UPDATE oder DELETE enthält. Der Server mit zusammengeschlossenen Datenbanken interpretiert bestimmte Aktionen ebenfalls als Aktualisierungen. Dazu gehören einige Transaktionen mit Anweisungen des Typs SELECT. Weitere Informationen hierzu finden Sie im Abschnitt "Was ist eine Aktualisierung in einem System zusammengeschlossener Datenbanken?".

Aktualisierungen können lokal oder fern ausgeführt werden. Lokale Aktualisierungen sind Aktualisierungen von Objekten in der zusammengeschlossenen Datenbank. Tabellen und Sichten in der zusammengeschlossenen Datenbank sind lokale Objekte. Ferne Aktualisierungen sind Aktualisierungen von Objekten in einer fernen Datenquelle.

Für Aktualisierungen von Kurznamen gelten folgende Einschränkungen:

- Ein Objekt, dessen Datenquelle keine Aktualisierungen zulässt, kann nicht aktualisiert werden.
- Ein Datenquellenobjekt, für das nur Lesezugriff besteht (z. B. die Sicht JOIN), kann nicht aktualisiert werden.
- Für Sichten für zusammengeschlossene Datenbanken, die mit der Anweisung UNION ALL erstellt wurden, können keine Operationen des Typs INSERT, UPDATE oder DELETE ausgeführt werden. Für Sichten für zusammengeschlossene Datenbanken, die mit der Anweisung UNION ALL erstellt wurden, besteht nur Lesezugriff.

Nicht unterstützte Datenquellen

Zusammengeschlossene Datenbanken unterstützen keine Operationen des Typs INSERT, UPDATE oder DELETE für die folgenden Datenquellen:

- BLAST
- BioRS

- Documentum
- Entrez
- Excel
- Extended Search
- HMMER
- Dateien mit Tabellenstruktur
- Web-Services
- WebSphere® Business Integration
- XML

Zugehörige Konzepte:

- „Zugriffsrechte für die Anweisungen INSERT, UPDATE und DELETE“ auf Seite 109
- „Referenzielle Integrität in einem System zusammengeschlossener Datenbanken“ auf Seite 111
- „Anweisungen INSERT, UPDATE und DELETE für große Objekte (LOBs)“ auf Seite 112

Referenzielle Integrität in einem System zusammengeschlossener Datenbanken

In einem System zusammengeschlossener Datenbanken kompensiert DB2® UDB keine Unterschiede bei der referenziellen Integrität zwischen Datenquellen. DB2 UDB hat keinen Einfluss auf das Erzwingen von referenzieller Integrität in den Datenquellen.

Referenzielle Integritätsbedingungen der Datenquelle können sich jedoch auf die Aktualisierung von Kurznamen auswirken. Angenommen, die zusammengeschlossene Datenbank empfängt eine Transaktion zum Einfügen von Daten in einen Kurznamen. Wenn der Server mit zusammengeschlossenen Datenbanken die Einfügung an die Datenquelle sendet, verletzt er damit eine referenzielle Integritätsbedingung in der Datenquelle. DB2 UDB ordnet den aufgetretenen Fehler einem DB2 UDB-Fehler zu.

Für die referenzielle Integrität zwischen Datenquellen sind die Anwendungen verantwortlich.

DB2 Information Integrator unterstützt informative Integritätsbedingungen, die Sie verwenden können, um Informationen zu referenziellen Integritätsbedingungen an den Abfrageprozessor zu übergeben.

Zugehörige Konzepte:

- „Informative Integritätsbedingungen für Kurznamen“ auf Seite 213
- „Interaktion von Clientanwendungen mit Datenquellen“ auf Seite 229
- „Gewährleisten der Anweisungsautarkie in einem System zusammengeschlossener Datenbanken“ auf Seite 112

Anweisungen INSERT, UPDATE und DELETE für große Objekte (LOBs)

Bei den drei LOB-Datentypen in DB2[®] UDB handelt es sich um große Zeichenobjekte (CLOBs = Character Large Objects), große Doppelbytezeichenobjekte (DBCLOBs = Double-Byte Character Large Objects) und große Binärobjekte (BLOBs = Binary Large Objects).

Bei einem Zusammenschluss von Datenbanken können in allen relationalen Datenquellen Lesevorgänge für LOBs ausgeführt werden. In Oracle-Datenquellen (Oracle ab Version 8), die den NET8-Wrapper verwenden, können Schreiboperationen für LOBs ausgeführt werden.

Zugehörige Konzepte:

- „LOB-Unterstützung in einem System zusammenschlossener Datenbanken“ auf Seite 237
- „LOB-Querverweise“ auf Seite 239
- „Einschränkungen für LOBs“ auf Seite 239

Zugehörige Referenzen:

- „Ändern langer Datentypen in VARCHAR-Datentypen“ auf Seite 64

Gewährleisten der Anweisungsautarkie in einem System zusammenschlossener Datenbanken

Während der Ausführung von Aktualisierungsoperationen versuchen Systeme zusammenschlossener Datenbanken immer, Daten nach Beendigung einer DML-Anweisung in einem autarken Status beizubehalten. Wenn sich Daten in einem autarken Status befinden, ist gewährleistet, dass sie entweder erfolgreich verarbeitet werden oder unverändert bleiben.

Wenn ein Client oder eine Anwendung eine Anweisung des Typs INSERT, UPDATE oder DELETE für einen Kurznamen absetzt, verarbeitet der Server mit zusammenschlossener Datenbanken diese Anweisung intern entweder als eine einzelne DML-Anweisung oder als eine Reihe von DML-Anweisungen. Muss ein Server mit zusammenschlossener Datenbanken mehrere DML-Anweisungen zur Verarbeitung an eine Zieldatenquelle weiterleiten, wird dadurch die Datenautarkie möglicherweise beeinträchtigt. Um eine Beeinträchtigung der Datenautarkie zu vermeiden, verwenden Systeme zusammenschlossener Datenbanken Sicherungspunkt-APIs für Datenquellen, um die Reihe von DML-Anweisungen zu überwachen.

Einige Datenquellen oder Versionen von Datenquellen lagern keine Sicherungspunkt-APIs aus, die das System zusammenschlossener Datenbanken verwenden kann. In solchen Fällen werden Anweisungen des Typs INSERT, UPDATE oder DELETE in Systemen zusammenschlossener Datenbanken ohne den Schutz von Sicherungspunkt-APIs ausgeführt.

Wenn während der Ausführung von Anweisungen des Typs INSERT, UPDATE oder DELETE in Systemen zusammenschlossener Datenbanken ein Fehler auftritt, kann es in der Datenquelle zu unvollständigen Aktualisierungen kommen. Zur Korrektur möglicher Inkonsistenzen führt ein System zusammenschlossener Datenbanken automatisch eine interne ROLLBACK-Operation für Transaktionen aus, bevor es einen SQLCODE-Fehler an die Anwendungen ausgibt.

Die folgenden Datenquellen lagern keine vom System zusammenschlossener Datenbanken verwendbaren Sicherungspunkt-APIs aus:

- DB2[®] für Linux, UNIX[®] und Windows[®], Version 6 (oder früher)
- DB2 für iSeries[™]
- DB2 für VM und VSE
- DB2 für z/OS[™] und OS/390[®], Version 5 (oder früher)
- Informix[®]
- Microsoft[®] SQL Server
- ODBC
- Teradata

Wenn eine vollständige INSERT-, UPDATE- oder DELETE-Transaktion im Push-down-Modus zur Verarbeitung an die Datenquelle weitergeleitet wird, nimmt der Server mit zusammenschlossener Datenbanken an, dass die Datenquelle die Anweisungsautarkie gewährleistet, falls ein Fehler auftritt.

Wenn nur ein Teil einer INSERT-, UPDATE- oder DELETE-Transaktion im Push-down-Modus zur Verarbeitung an die Datenquelle weitergeleitet wird, führt das System bei Auftreten eines Fehlers eine ROLLBACK-Operation aus, um die gesamte Transaktion rückgängig zu machen.

Dieses Verhalten wird von der Serveroption IUD_APP_SVPT_ENFORCE gesteuert. Diese Serveroption ist standardmäßig auf 'Y' gesetzt, sodass Transaktionen, bei denen ein Fehler auftritt, mit einer ROLLBACK-Operation rückgängig gemacht werden.

Wenn Sie die Anweisung SET SERVER OPTION verwenden, um die Serveroption IUD_APP_SVPT_ENFORCE für die Dauer einer Verbindung festzulegen, hat dies keine Auswirkungen auf statische SQL-Anweisungen. Wenn Sie diese Serveroption mit der Anweisung SET SERVER OPTION festlegen, wirkt sich die Option nur auf dynamische SQL-Anweisungen aus.

Szenario für das Verhalten der Serveroption IUD_APP_SVPT_ENFORCE:

Angenommen, Sie erstellen dem Kurznamen INFMX_UT für eine Informix-Tabelle mit dem Namen UT. Die Tabelle UT enthält vier Spalten für ganzzahlige Daten, i1, i2, i3 und i4. Die Spalte i1 ist eine eindeutige Indexspalte.

Die Tabelle UT ist leer. Sie setzen die Anweisung INSERT für den Kurznamen INF_X_UT ab, um die Werte 1, 22, 34 und 40 in Zeile 1 der Tabelle einzufügen. Die Anweisung ist erfolgreich.

Anschließend setzen Sie eine Anweisung INSERT für mehrere Zeilen für den Kurznamen INF_X_UT ab, um drei Datenzeilen einzufügen:

- Zeile 2: 2, 37, 34, 55
- Zeile 3: 3, 42, 59, 40
- Zeile 4: 1, 55, 62, 75

Da die Daten, die in der letzten Zeile eingefügt werden sollen, die Bestimmung verletzen, dass Spalte i1 ein eindeutiger Index sein soll, gibt der Informix-Server eine Fehlermeldung an das System zusammenschlossener Datenbanken zurück. Das System zusammenschlossener Datenbanken gibt den SQL-Fehler SQL0803N

an Ihre Anwendung zurück. Die Fehlermeldung SQL0803N beschreibt die Nichteinhaltung der Bestimmung für eindeutige Indizes.

In der folgenden Tabelle werden die Zeilen der Tabelle UT nach den INSERT-Operationen aufgeführt:

Tabelle 8. Beispiel: Informix-Tabelle UT

	Spalte (eindeutiger Index)	Spalte	Spalte	Spalte
Zeile	i1	i2	i3	i4
Zeile 1	1	22	34	40
Zeile 2	2	37	34	55
Zeile 3	3	42	59	40

Standardverhalten der Serveroption IUD_APP_SVPT_ENFORCE:

Die Serveroption IUD_APP_SVPT_ENFORCE ist standardmäßig auf 'Y' gesetzt. Wenn diese Serveroption auf 'Y' gesetzt ist, führt sie intern eine ROLLBACK-Operation für die gesamte Transaktion aus. Obwohl die ersten beiden Datenzeilen erfolgreich eingefügt wurden, wird die Einfügung rückgängig gemacht, da die ROLLBACK-Operation für die gesamte Transaktion ausgeführt wird.

Wenn Sie das Standardverhalten der Serveroption IUD_APP_SVPT_ENFORCE ändern wollen, verwenden Sie die Anweisung ALTER SERVER. Diese Änderung wird auf alle Datenquellenobjekte angewendet, auf die über die von Ihnen angegebenen Server zugegriffen wird.

Alternatives Verhalten der Serveroption IUD_APP_SVPT_ENFORCE:

Wenn die Serveroption IUD_APP_SVPT_ENFORCE auf 'N' gesetzt ist, wird die Transaktion nicht rückgängig gemacht. Die zweite und dritte Datenspalte werden nicht aus der Tabelle entfernt. Die Fehlerbehebung muss von Ihrer Anwendung übernommen werden.

Zugehörige Konzepte:

- „Referenzielle Integrität in einem System zusammenschlossener Datenbanken“ auf Seite 111

Zugehörige Referenzen:

- Kapitel 21, „Serveroptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 253

Arbeiten mit Kurznamen

Wenn Sie Daten der Datenquelle auswählen oder ändern wollen, führen Sie unter Verwendung der Anweisungen SELECT, INSERT, UPDATE und DELETE Abfragen für die Kurznamen aus. Sie übergeben Abfragen in DB2-SQL an die zusammengeslossene Datenbank. Sie können Daten aus lokalen Tabellen und Daten aus ferneren Datenquellen unter Verwendung einer einzigen SQL-Anweisung so miteinander verknüpfen, als würde es sich nur um lokale Daten handeln. Sie können z. B. Daten miteinander verknüpfen, die sich an folgenden Speicherpositionen befinden:

- In einer lokalen Tabelle von DB2 für Windows in der zusammengeschlossenen Datenbank, in einer Oracle-Tabelle und in einer Sybase-Sicht
- In einer Tabelle von DB2 UDB für z/OS auf einem Server, in einer Tabelle von DB2 UDB für z/OS auf einem anderen Server und in einem Excel-Arbeitsblatt

Indem das System zusammengeschlossener Datenbanken SQL-Anweisungen so verarbeitet, als wären die Datenquellen normale relationale Tabellen oder Sichten innerhalb der zusammengeschlossenen Datenbank, kann das System relationale Daten und nicht relationale Daten miteinander verknüpfen.

Tabellen und Sichten in der zusammengeschlossenen Datenbank sind *lokale Objekte*. Geben Sie diesen Objekten keine Kurznamen, sondern verwenden Sie in Anweisungen die eigentlichen Objektnamen.

Ferne Objekte sind Objekte, die sich nicht in der zusammengeschlossenen Datenbank befinden. Für diese Objekte müssen Kurznamen erstellt werden. Beispiele:

- Tabellen und Sichten in einer anderen Datenbank oder einem anderen Exemplar von DB2 für Linux, UNIX und Windows auf dem System zusammengeschlossener Datenbanken
- Tabellen und Sichten in einer anderen Datenbank oder einem anderen Exemplar von DB2 für Linux, UNIX und Windows auf einem anderen System
- Objekte, die sich in anderen Datenquellen als DB2 für Linux, UNIX und Windows befinden, z. B. in: Oracle, Sybase, Documentum und ODBC

Vorgehensweise:

Um die Vorteile eines Systems zusammengeschlossener Datenbanken zu nutzen, müssen Sie über Kenntnisse in folgenden Bereichen verfügen:

- SQL-Anweisungen, die mit Kurznamen verwendet werden können
- Zugriff auf neue Datenquellenobjekte
- Verwendung einer Durchgriffssitzung für den direkten Zugriff auf Datenquellen
- Vorteile des Einsatzes von Sichten zusammengeschlossener Datenbanken beim Zugriff auf heterogene Daten

WITH HOLD-Syntax

Sie können die WITH HOLD-Syntax für einen Cursor verwenden, der für einen Kurznamen definiert ist. Sie empfangen allerdings einen Fehler, wenn Sie versuchen, diese Syntax (mit einer COMMIT-Operation) zu verwenden, und die Datenquelle die WITH HOLD-Syntax nicht unterstützt.

Auslöser

Ein Kurzname kann kein Aktualisierungsziel in einem Auslöser sein. Sie können Anweisungen des Typs SELECT in den Auslöserhauptteil aufnehmen. Sie können in den Auslöserhauptteil keine Anweisungen des Typs INSERT, UPDATE oder DELETE für den Kurznamen aufnehmen.

Zugehörige Konzepte:

- „Zugriffsrechte für die Anweisungen INSERT, UPDATE und DELETE“ auf Seite 109
- „Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammengeschlossener Datenbanken“ auf Seite 110

Zugehörige Tasks:

- „Verweisen auf Datenquellenobjekte in SQL-Anweisungen über Kurznamen“ auf Seite 230
- „Mit Kurznamen verwendbare SQL-Anweisungen“ auf Seite 116
- „Zugreifen auf neue Datenquellenobjekte“ auf Seite 121
- „Zugreifen auf Datenquellen mit Durchgriffssitzungen“ auf Seite 123
- „Zugreifen auf heterogene Daten durch Sichten für zusammengeschlossene Datenbanken“ auf Seite 124

Funktionsweise von Kurznamen - ausführliche Informationen

In diesem Abschnitt werden die INSERT-, UPDATE- und DELETE-Operationen für Kurznamen beschrieben.

Mit Kurznamen verwendbare SQL-Anweisungen

Ein System zusammengeschlossener Datenbanken ermöglicht den einfachen Zugriff auf Daten unabhängig davon, wo diese gespeichert sind. Um auf die Daten zuzugreifen, müssen Sie Kurznamen für alle Datenquellenobjekte (z. B. Tabellen und Sichten) erstellen, die sie abrufen wollen.

Wird beispielsweise der Kurzname DEPT für die Darstellung der fernen Tabelle EUROPE.PERSON.DEPT erstellt, können Sie die Anweisung `SELECT * FROM DEPT` verwenden, um die Daten in der fernen Tabelle abzufragen. Sie führen die Abfrage für den Kurznamen aus und müssen sich so nicht die zugrunde liegenden Informationen der Datenquelle merken. Beim Erstellen einer Abfrage müssen Sie z. B. Folgendes nicht berücksichtigen:

- Den Namen des Objekts in der Datenquelle
- Den Server, auf dem sich die Datenquellenobjekte befinden
- Den Typ der Datenquelle, in der sich das Objekt befindet, z. B. Informix oder Oracle
- Die von der Datenquelle verwendete Abfragesprache bzw. SQL-Programmversion
- Die Datentypzuordnungen zwischen der Datenquelle und DB2 Information Integrator
- Die Funktionszuordnungen zwischen der Datenquelle und DB2 Information Integrator

Die zugrunde liegenden Metadaten, die im Katalog der zusammengeschlossenen Datenbank gespeichert sind, stellen dem Server mit zusammengeschlossenen Datenbanken die Informationen zur Verfügung, die dieser zur Verarbeitung von Abfragen benötigt. Diese Metadaten werden aus den Datenquellen zusammengestellt, wenn der Server mit zusammengeschlossenen Datenbanken und die zugehörige Datenbank installiert und für den Zugriff auf die Datenquellen konfiguriert werden.

Nachdem das System zusammengeschlossener Datenbanken eingerichtet ist, können Sie die Kurznamen für Abfragen der Datenquellen oder zum weiteren Ausbau der Konfiguration des Systems zusammengeschlossener Datenbanken verwenden.

In der folgenden Tabelle sind die SQL-Anweisungen aufgelistet, die die Verwendung von Kurznamen unterstützen:

Tabelle 9. Allgemeine SQL-Anweisungen, die die Verwendung von Kurznamen unterstützen

SQL-Anweisung	Beschreibung	Erforderliche Berechtigung
ALTER NICKNAME	Ändert einen vorhandenen Kurznamen durch Änderung des lokalen Spaltennamens, des lokalen Datentyps, der Spaltenoptionen der zusammengesetzten Datenbank oder der informativen Integritätsbedingungen. Die Tabelle oder Sicht in der Datenquelle wird davon nicht beeinflusst.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht ALTER oder CONTROL für den Kurznamen • 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
ALTER TABLE	Ändert eine ferne Tabelle, die mit DB2 Information Integrator unter Verwendung transparenter DDL erstellt wurde. Sie können keine Tabellen ändern, die nativ in den Datenquellen erstellt wurden. Kann informative Integritätsbedingungen verwenden, um referenzielle Integritätsbedingungen zu einem Kurznamen hinzuzufügen.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht ALTER oder CONTROL für den Kurznamen • Zugriffsrecht ALTERIN für das Schema, wenn der Schemaname des Kurznamens vorhanden ist
COMMENT ON	Ersetzt Kommentare oder fügt Kommentare in den Katalogbeschreibungen verschiedener Objekte hinzu. Hierzu gehören Funktionen, Funktionszuordnungen, Indizes, Kurznamen, Server, Serveroptionen, Typzuordnungen und Wrapper.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht ALTER oder CONTROL für das Objekt • Zugriffsrecht ALTERIN für das Schema • Definierender Benutzer des Objekts, der in der Spalte DEFINER der Katalogsicht für das Objekt aufgezeichnet wurde
CREATE ALIAS	Definiert einen Aliasnamen für einen Kurznamen.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Berechtigung IMPLICIT_SCHEMA für die Datenbank, wenn der implizite oder der explizite Schemaname des Aliasnamens nicht vorhanden ist • Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname des Aliasnamens auf ein bestehendes Schema bezieht
CREATE INDEX (mit Klausel SPECIFICATION ONLY)	Erstellt eine Indexspezifikation (Metadaten), die dem Abfrageoptimierungsprogramm anzeigt, dass ein Datenquellenobjekt einen Index hat. Es wird nicht der Index selbst, sondern nur die Spezifikation erstellt.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht CONTROL oder INDEX für das zugrunde liegende Datenquellenobjekt — <i>und entweder</i> die Berechtigung IMPLICIT_SCHEMA für die Datenbank oder die Berechtigung CREATEIN für das Schema

Tabelle 9. Allgemeine SQL-Anweisungen, die die Verwendung von Kurznamen unterstützen (Forts.)

SQL-Anweisung	Beschreibung	Erforderliche Berechtigung
CREATE TABLE (mit Klausel OPTI- ONS)	Erstellt mit DB2 Information Integrator eine ferne Tabelle unter Verwendung von transparenter DDL.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht CREATETAB für die Datenbank und Zugriffsrecht USE für den Tabellenbereich sowie <i>entweder</i> die Berechtigung IMPLICIT_SCHEMA für die Datenbank <i>oder</i> das Zugriffsrecht CREATEIN für das Schema
CREATE TABLE (mit der Gesamtauswahl AS und den Klauseln DATA INITIALLY DEFERRED REFRESH)	Erstellt eine gespeicherte Abfragetabelle unter Verwendung einer Gesamtauswahl, die einen Kurznamen verwendet	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht CREATETAB für die Datenbank und Zugriffsrecht USE für den Tabellenbereich sowie <i>entweder</i> die Berechtigung IMPLICIT_SCHEMA für die Datenbank <i>oder</i> das Zugriffsrecht CREATEIN für das Schema • Zugriffsrecht CONTROL für die Tabelle oder Sicht • Zugriffsrecht SELECT für die Tabelle oder Sicht und Zugriffsrecht ALTER, wenn REFRESH DEFERRED angegeben wird
CREATE VIEW	Erstellt eine Sicht, die auf einen oder mehrere Kurznamen verweist.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht CONTROL oder SELECT für den Kurznamen und <i>entweder</i> die Berechtigung IMPLICIT_SCHEMA für die Datenbank <i>oder</i> das Zugriffsrecht CREATEIN für das Schema
DELETE	Löscht Zeilen aus einem Datenquellenobjekt (z. B. einer Tabelle oder einer Sicht), für das ein Kurzname erstellt wurde.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht DELETE für den Kurznamen und Zugriffsrecht DELETE für das zugrunde liegende Datenquellenobjekt • Zugriffsrecht CONTROL für das zugrunde liegende Datenquellenobjekt

Tabelle 9. Allgemeine SQL-Anweisungen, die die Verwendung von Kurznamen unterstützen (Forts.)

SQL-Anweisung	Beschreibung	Erforderliche Berechtigung
DROP	<p>Löscht ein Objekt, z. B. einen Kurznamen, eine Sicht für zusammengesessene Datenbanken oder eine Indexspezifikation. Die Tabelle, Sicht oder der Index in der Datenquelle wird davon nicht beeinflusst.</p> <p>Wenn Tabellen gelöscht werden, die mit transparenter DDL erstellt wurden, werden die zugehörigen Kurznamen für diese Tabellen ebenfalls gelöscht.</p>	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht DROPIN für das Schema des Objekts • Zugriffsrecht CONTROL für das Objekt
GRANT	<p>Erteilt Zugriffsrechte für Kurznamen und Sichten für zusammengesessene Datenbanken, wie z. B. ALTER, DELETE, INDEX, INSERT, SELECT, UPDATE. Zugriffsrechte für die Datenquelle müssen separat erteilt werden.</p>	<ul style="list-style-type: none"> • SYSADM oder DBADM • WITH GRANT OPTION für jedes angegebene Zugriffsrecht • Zugriffsrecht CONTROL für das Objekt
INSERT	<p>Fügt Zeilen in das Datenquellenobjekt (z. B. einer Tabelle oder einer Sicht) ein, für das ein Kurzname erstellt wurde.</p>	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht INSERT für den Kurznamen und Zugriffsrecht INSERT für das zugrunde liegende Datenquellenobjekt • Zugriffsrecht CONTROL für das zugrunde liegende Datenquellenobjekt
LOCK TABLE	<p>Löst das Sperren des fernen Objekts in der Datenquelle aus. Verhindert, dass gleichzeitig ablaufende Anwendungsprozesse eine Datenquellentabelle ändern, die über einen Kurznamen verfügt. Diese Anweisung wird für ODBC, Microsoft SQL Server und nicht relationale Datenquellen nicht unterstützt.</p>	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht SELECT für die zugrunde liegende Tabelle • Zugriffsrecht CONTROL für die zugrunde liegende Tabelle
REVOKE	<p>Widerruft Zugriffsrechte für Kurznamen und Sichten für zusammengesessene Datenbanken, wie z. B. ALTER, DELETE, INDEX, INSERT, SELECT, UPDATE. Zugriffsrechte für die Datenquelle müssen separat widerrufen werden.</p>	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht CONTROL für das Objekt

Tabelle 9. Allgemeine SQL-Anweisungen, die die Verwendung von Kurznamen unterstützen (Forts.)

SQL-Anweisung	Beschreibung	Erforderliche Berechtigung
SELECT	Wählt Zeilen aus dem Datenquellenobjekt (z. B. aus einer Tabelle oder einer Sicht) aus, für das ein Kurzname erstellt wurde.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht SELECT für den Kurznamen und Zugriffsrecht SELECT für das zugrunde liegende Datenquellenobjekt • Zugriffsrecht CONTROL für das zugrunde liegende Datenquellenobjekt
UPDATE	Aktualisiert die Werte in angegebenen Zeilenspalten von Datenquellenobjekten (z. B. Tabellen oder Sichten), für die ein Kurzname erstellt wurde.	<ul style="list-style-type: none"> • SYSADM oder DBADM • Zugriffsrecht UPDATE für den Kurznamen und Zugriffsrecht UPDATE für das zugrunde liegende Datenquellenobjekt • Zugriffsrecht CONTROL für das zugrunde liegende Datenquellenobjekt

Wird eine Abfrage an die zusammengeschlossene Datenbank übergeben, werden die Zugriffsrechte für den Kurznamen in der Abfrage überprüft. Die Berechtigungsanforderungen des Datenquellenobjekts, auf das der Kurzname verweist, werden nur angewendet, wenn die Abfrage tatsächlich in der Datenquelle verarbeitet wird.

Um Daten unter Verwendung eines Kurznamens auszuwählen, einzufügen, zu aktualisieren oder zu löschen, muss die Berechtigungs-ID der Anweisung über folgende Zugriffsrechte verfügen:

- Das entsprechende Zugriffsrecht für den Kurznamen (sodass die zusammengeschlossene Datenbank die Anforderung annimmt)
- Das entsprechende Zugriffsrecht für das zugrunde liegende Tabellenobjekt (sodass die zusammengeschlossene Datenbank die Anforderung annimmt)

Um beispielsweise eine Datenquelle unter Verwendung eines Kurznamens zu aktualisieren, benötigen Sie das Zugriffsrecht UPDATE für den Kurznamen und das Zugriffsrecht UPDATE für das zugrunde liegende Datenquellenobjekt.

Zugehörige Tasks:

- „Verweisen auf Datenquellenobjekte in SQL-Anweisungen über Kurznamen“ auf Seite 230

Zugehörige Referenzen:

- „ALTER TABLE statement“ in *SQL Reference, Volume 2*
- „COMMENT statement“ in *SQL Reference, Volume 2*
- „CREATE ALIAS statement“ in *SQL Reference, Volume 2*
- „CREATE INDEX statement“ in *SQL Reference, Volume 2*
- „CREATE VIEW statement“ in *SQL Reference, Volume 2*
- „DELETE statement“ in *SQL Reference, Volume 2*
- „DROP statement“ in *SQL Reference, Volume 2*
- „GRANT (Database Authorities) statement“ in *SQL Reference, Volume 2*
- „INSERT statement“ in *SQL Reference, Volume 2*
- „LOCK TABLE statement“ in *SQL Reference, Volume 2*

- „REVOKE (Table, View, or Nickname Privileges) statement“ in *SQL Reference, Volume 2*
- „SELECT statement“ in *SQL Reference, Volume 2*
- „UPDATE statement“ in *SQL Reference, Volume 2*
- „ALTER NICKNAME statement“ in *SQL Reference, Volume 2*

Zugreifen auf neue Datenquellenobjekte

Sie möchten möglicherweise in regelmäßigen Abständen auf Datenquellenobjekte zugreifen, die keine Kurznamen haben. Dies können zu der Datenquelle neu hinzugefügte Objekte sein, wie z. B. eine neu erstellte Sicht. Oder dies können bereits vorhandene Objekte sein, die auf dem Server mit zusammengeschlossenen Datenbanken bei der anfänglichen Einrichtung nicht registriert wurden. In beiden Fällen handelt es sich um für den Server mit zusammengeschlossenen Datenbanken neue Objekte. Um auf diese neuen Objekte zugreifen zu können, müssen mit der Anweisung CREATE NICKNAME Kurznamen dafür erstellt werden.

Voraussetzungen:

Das System zusammengeschlossener Datenbanken muss für den Zugriff auf die Datenquelle konfiguriert sein.

In der zusammengeschlossenen Datenbank muss eine Serverdefinition für den Datenquellenserver, auf dem sich das Objekt befindet, vorhanden sein. Eine Serverdefinition wird unter Verwendung der Anweisung CREATE SERVER erstellt.

Wenn Sie Daten unter Verwendung eines Kurznamens einfügen, aktualisieren oder löschen wollen, müssen alle folgenden Bedingungen für Zugriffsrechte erfüllt sein:

- Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen die erforderlichen Zugriffsrechte SELECT, INSERT, UPDATE, und DELETE für den Kurznamen umfassen, damit die zusammengeschlossene Datenbank die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss über die erforderlichen Zugriffsrechte SELECT, INSERT, UPDATE und DELETE für das zugrunde liegende Tabellenobjekt verfügen, damit die Datenquelle die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss mit einer Benutzerzuordnung der Berechtigungs-ID auf dem Server mit zusammengeschlossenen Datenbanken zugeordnet werden.

Zum Ausgeben der Anweisung CREATE NICKNAME müssen Sie über eine der folgenden Berechtigungen verfügen:

- SYSADM oder DBADM
- Berechtigung IMPLICIT_SCHEMA für die zusammengeschlossene Datenbank, wenn der implizite oder der explizite Schemaname des Kurznamens nicht vorhanden ist.
- Zugriffsrecht CREATEIN für das Schema, wenn der Schemaname des Kurznamens vorhanden ist.

Vorgehensweise:

Die Anweisung CREATE NICKNAME weist für relationale und nicht relationale Datenquellen leichte Unterschiede auf.

Für relationale Datenquellen lautet die Syntax der Anweisung CREATE NICKNAME wie folgt:

```
CREATE NICKNAME nickname_name FOR server_name."remote_schema".object_name
  OPTIONS (options_list)
```

nickname_name

Ein eindeutiger Kurzname für das Datenquellenobjekt.

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, das Schema des Kurznamens. Kurznamen können bis zu 128 Zeichen lang sein.

FOR *server_name*."*remote_schema*".*object_name*"

Eine aus drei Teilen bestehende Kennung für das ferne Datenquellenobjekt. Wenn Ihre Datenquelle keine Schemas unterstützt, lassen Sie das Schema in der Anweisung CREATE NICKNAME weg.

- *server_name* ist der Name, der in der Anweisung CREATE SERVER dem Datenquellenserver zugeordnet wurde.
- *remote_schema* ist der Name des fernen Schemas, zu dem das Objekt gehört.
- *object_name* ist der Name des fernen Objekts, auf das zugegriffen werden soll.

OPTIONS (*options_list*)

Informationen zum Kurznamen, die dem SQL-Abfrage-Compiler und dem Wrapper das effiziente Ausführen von Abfragen ermöglichen.

Für einige nicht relationale Datenquellen lautet die Syntax der Anweisung CREATE NICKNAME wie folgt:

```
CREATE NICKNAME nickname_name column_definition_list
  FOR SERVER server_name
  OPTIONS (options_list)
```

nickname_name

Ein eindeutiger Kurzname für das Datenquellenobjekt.

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, das Schema des Kurznamens. Kurznamen können bis zu 128 Zeichen lang sein.

column_definition_list

Eine Liste mit Kurznamenspalten und Datentypen.

FOR SERVER *server_name*

Der lokale Name, den Sie für den fernen Server mit der Anweisung CREATE SERVER in den Serverdefinitionsinformationen erstellt haben.

OPTIONS (*options_list*)

Informationen zum Kurznamen, die dem SQL-Abfrage-Compiler und dem Wrapper das effiziente Ausführen von Abfragen ermöglichen.

Zugehörige Tasks:

- „Arbeiten mit Kurznamen“ auf Seite 114

Zugehörige Referenzen:

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

Zugreifen auf Datenquellen mit 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-SQL oder -API nicht unterstützt wird. Verwenden Sie beispielsweise eine Durchgriffssitzung, um unter Verwendung der nativen Programmversion der Datenquelle eine Prozedur oder einen Index zu erstellen oder Abfragen auszuführen.

Momentan akzeptieren die Datenquellen, die die Durchgriffsfunktion unterstützen, SQL-Anweisungen nur in einer Durchgriffssitzung. 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 Verwaltungstasks. Sie können jedoch nicht alle Verwaltungstasks mit Hilfe einer Durchgriffssitzung durchführen. Die ausführbaren Verwaltungstasks sind von der Datenquelle abhängig. Bei DB2 UDB können Sie beispielsweise das von der Datenquelle verwendete Statistikdienstprogramm ausführen, Sie können jedoch die ferne Datenbank nicht starten oder stoppen.

In einer Durchgriffssitzung können Sie nur eine einzelne Datenquelle abfragen. Verwenden Sie den Befehl SET PASSTHRU, um eine Sitzung zu öffnen. Mit dem Befehl SET PASSTHRU RESET wird die Durchgriffssitzung geschlossen. Verwenden Sie den Befehl SET PASSTHRU an Stelle von SET PASSTHRU RESET, wird die aktuelle Durchgriffssitzung geschlossen und eine neue geöffnet.

Sie können die WITH HOLD-Syntax für einen Cursor verwenden, der in einer Durchgriffssitzung definiert wurde. Sie empfangen allerdings einen Fehler, wenn Sie versuchen, diese Syntax (mit einer COMMIT-Operation) zu verwenden, und die Datenquelle die WITH HOLD-Syntax nicht unterstützt.

Durchgriffssitzungen unterstützen keine nicht relationalen Datenquellen.

Zugehörige Konzepte:

- „Durchgriffssitzungen“ auf Seite 12
- „Direktes Abfragen von Datenquellen im Durchgriffsmodus“ auf Seite 242

Zugehörige Tasks:

- „Arbeiten mit Kurznamen“ auf Seite 114

Zugehörige Referenzen:

- „SET PASSTHRU statement“ in *SQL Reference, Volume 2*

Zugreifen auf heterogene Daten durch Sichten für zusammengeslossene Datenbanken

Eine *Sicht für zusammengeslossene Datenbanken* ist eine Sicht in der zusammengeslossenen Datenbank, deren Basistabellen sich in fernen Datenquellen befinden. Auf diese Basistabellen wird in der Sicht für zusammengeslossene Datenbanken durch Kurznamen und nicht durch die Tabellennamen der Datenquelle verwiesen. Wenn Sie eine Sicht für zusammengeslossene Datenbanken abfragen, werden die Daten von der fernen Datenquelle abgerufen. Die Aktion des Erstellens einer Sicht für zusammengeslossene Datenbanken für Datenquellendaten wird auch als das „Erstellen einer Sicht für einen Kurznamen“ bezeichnet. Der Grund dafür ist, dass Sie beim Erstellen der Sicht auf die Kurznamen und nicht auf die Datenquellen verweisen.

Diese Sichten bieten ein hohes Maß an Datenunabhängigkeit für eine global integrierte Datenbank, wie dies auch Sichten, die für mehrere lokale Tabellen definiert sind, für zentralisierte Manager relationaler Datenbanken tun.

Voraussetzungen:

Zum Ausgeben der Anweisung CREATE VIEW müssen Sie über eine der folgenden Berechtigungen verfügen:

- SYSADM oder DBADM
- Für jeden Kurznamen in jeder Gesamtauswahl die beiden folgenden Berechtigungen:
 - Zugriffsrecht CONTROL oder SELECT für die zugrunde liegende Tabelle oder Sicht
 - Eine der beiden folgenden Berechtigungen oder Zugriffsrechte:
 - Berechtigung IMPLICIT_SCHEMA für die zusammengeslossene Datenbank, wenn der implizite oder der explizite Schemaname der Sicht nicht vorhanden ist
 - Zugriffsrecht CREATEIN für das Schema, wenn sich der Schemaname der Sicht auf ein bestehendes Schema bezieht

Zugriffsrechte für zugrunde liegende Objekte werden beim Definieren einer Sicht für einen Kurznamen einer zusammengeslossenen Datenbank nicht in Betracht gezogen.

Einschränkungen:

Auf Sichten für zusammengeslossene Datenbanken, für die UNION ALL-Anweisungen definiert sind, besteht nur Lesezugriff.

Auf Sichten für zusammengeslossene Datenbanken, die mehr als einen Kurznamen in die Klausel FROM einschließen, besteht nur Lesezugriff.

Auf Sichten für zusammengeslossene Datenbanken, die nur einen Kurznamen in die Klausel FROM einschließen, besteht möglicherweise nur Lesezugriff.

- Wenn sich der Kurzname in der Klausel FROM auf eine nicht relationale Datenquelle bezieht, besteht auf die Sicht für zusammengeslossene Datenbanken nur Lesezugriff.

- Wenn Sie beim Erstellen der Sicht andere Kurznamen als Vergleichselemente oder Unterabfragen mit einschließen, kann die Sicht für zusammengeschlossene Datenbanken aktualisiert werden.

Vorgehensweise:

Verwenden Sie die Anweisung `CREATE VIEW`, um eine Sicht für zusammengeschlossene Datenbanken zu erstellen.

Berechtigungsanforderungen der Datenquelle für die Tabelle oder Sicht, auf die durch den Kurznamen verwiesen wird, werden bei Verarbeitung der Abfrage angewendet. Die Berechtigungs-ID der Anweisung kann einer anderen fernen Berechtigungs-ID durch eine Benutzerzuordnung zugeordnet werden.

Sicht für zusammengeschlossene Datenbanken mit Mischungen ähnlicher Daten aus verschiedenen Datenquellenobjekten erstellen:

Angenommen, Sie verfügen über Kundendaten auf drei verschiedenen Servern. Ein Server befindet sich in Europa, einer in Asien und einer in Südamerika. Die Kundendaten für Europa befinden sich in einer Oracle-Tabelle. Der Kurzname für diese Tabelle lautet `ORA_EU_CUST`. Die Kundendaten für Asien befinden sich in einer Sybase-Tabelle. Der Kurzname für diese Tabelle lautet `SYB_AS_CUST`. Die Kundendaten für Südamerika befinden sich in einer Informix-Tabelle. Der Kurzname für diese Tabelle lautet `INFMX_SA_CUST`. Jede Tabelle hat Spalten für die Kundennummer (`CUST_NO`), den Kundennamen (`CUST_NAME`), die Produktnummer (`PROD_NO`) und die Bestellmenge (`QUANTITY`). Die Syntax zum Erstellen einer Sicht von diesen drei Kurznamen, in der diese Kundendaten zusammengefügt werden, lautet wie folgt:

```
CREATE VIEW FV1
AS SELECT * FROM ORA_EU_CUST
UNION ALL
SELECT * FROM SYB_AS_CUST
UNION ALL
SELECT * FROM INFMX_SA_CUST
```

Daten zum Erstellen einer Sicht für zusammengeschlossene Datenbanken verknüpfen:

Angenommen, Sie haben auf einem Server Kundendaten und auf einem anderen Server Verkaufsdaten gespeichert. Die Kundendaten befinden sich in einer Oracle-Tabelle. Der Kurzname für diese Tabelle lautet `ORA_EU_CUST`. Die Verkaufsdaten befinden sich in einer Sybase-Tabelle. Der Kurzname für diese Tabelle lautet `SYB_SALES`. Sie möchten die Kundendaten mit den Käufen dieser Kunden abgleichen. Jede Tabelle hat eine Spalte für die Kundennummer (`CUST_NO`). Die Syntax zum Erstellen einer Sicht für zusammengeschlossene Datenbanken von diesen zwei Kurznamen, die diese Daten miteinander verknüpft, lautet wie folgt:

```
CREATE VIEW FV4
AS SELECT A.CUST_NO, A.CUST_NAME, B.PROD_NO, B.QUANTITY
FROM ORA_EU_CUST A, SYB_SALES B
WHERE A.CUST_NO=B.CUST_NO
```

Zugehörige Tasks:

- „Erstellen und Verwenden von Sichten für zusammengeslossene Datenbanken“ auf Seite 234
- „Arbeiten mit Kurznamen“ auf Seite 114

Zugehörige Referenzen:

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

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

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

Auswählen von Daten in einem System zusammengeslossener Datenbanken

Einige Typen verteilter Anforderungen, die in einem System zusammengeslossener Datenbanken verwendet werden, fragen folgende (Kombinationen von) Datenquellen ab:

- Eine einzelne ferne Datenquelle.
- Eine lokale und eine ferne Datenquelle.
- Mehrere ferne Datenquellen.
- Eine Kombination von fernen und lokalen Datenquellen.

Um Daten in Datenquellen auszuwählen, müssen Sie in der Anweisung SELECT die Kurznamen der Datenquellenobjekte verwenden.

Voraussetzungen:

Wenn Sie Daten unter Verwendung eines Kurznamens auswählen wollen, müssen alle folgenden Bedingungen für Zugriffsrechte erfüllt sein:

- Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen das Zugriffsrecht SELECT für den Kurznamen umfassen, damit die zusammengeschlossene Datenbank die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss über das Zugriffsrecht SELECT für das zugrunde liegende Tabellenobjekt verfügen, damit die Datenquelle die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss mit einer Benutzerzuordnung der Berechtigungs-ID auf dem Server mit zusammengeschlossenen Datenbanken zugeordnet werden.

Vorgehensweise:

Die zusammengeschlossene Datenbank ist eine lokale Datenquelle. Tabellen und Sichten in der zusammengeschlossenen Datenbank sind lokale Objekte. Für diese Objekte erstellen Sie keine Kurznamen, sondern Sie verwenden in der Anweisung SELECT den eigentlichen Objektnamen.

Zu fernen Datenquellen gehören: ein weiteres Datenbankexemplar von DB2 für Linux, UNIX und Windows auf dem Server mit zusammengeschlossenen Datenbanken, ein weiteres Datenbankexemplar von DB2 für Linux, UNIX und Windows auf einem anderen Server und Datenquellen, die keine Datenquellen von DB2 für Linux, UNIX und Windows sind. Objekte, die in fernen Datenquellen gespeichert sind, werden als ferne Objekte bezeichnet.

Angenommen, ein Server mit zusammengeschlossenen Datenbanken ist für den Zugriff auf eine Datenquelle von DB2 für OS/390, eine Datenquelle von DB2 für iSeries und eine Datenquelle von Oracle konfiguriert. In jeder dieser Datenquellen ist eine Tabelle gespeichert, die Verkaufsdaten enthält. Diese Konfiguration wird in der folgenden Abbildung dargestellt.

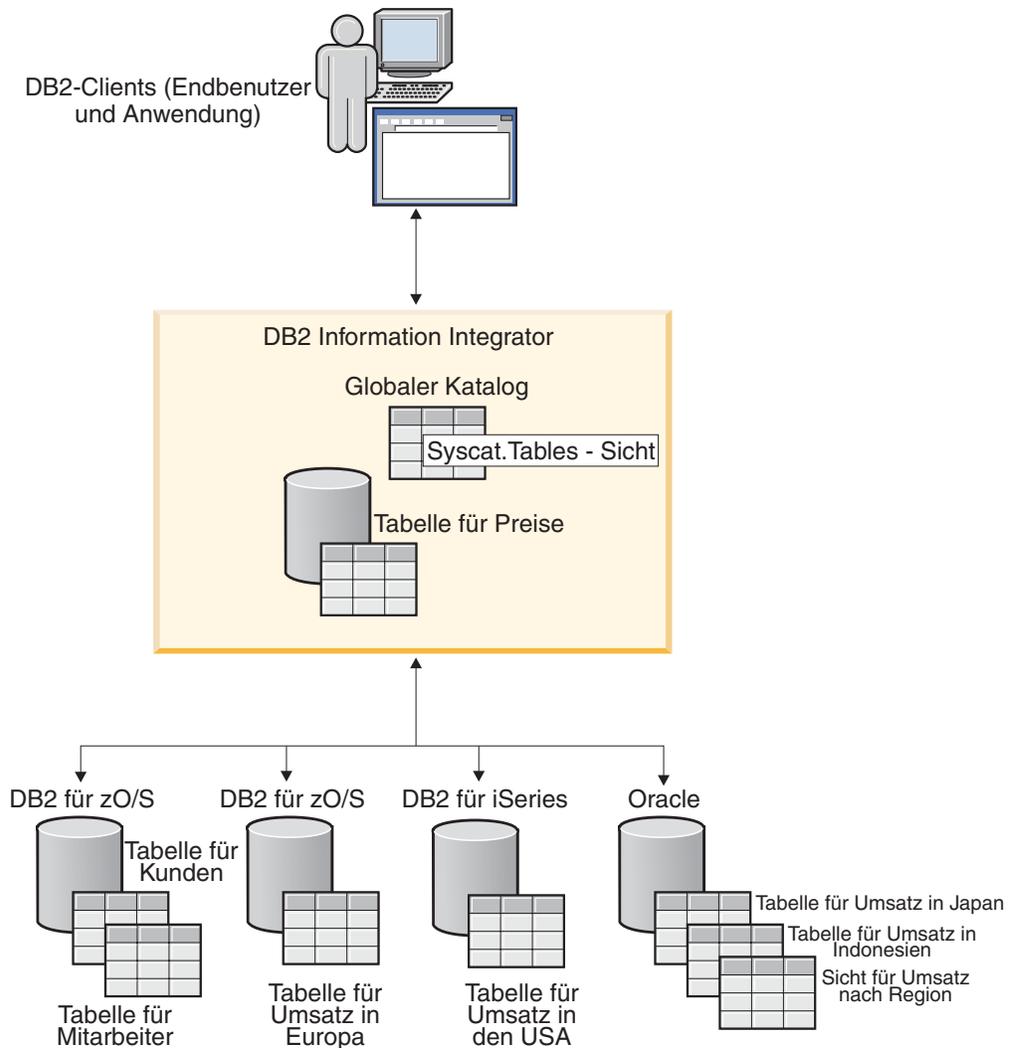


Abbildung 3. Beispiel für ein System zusammengeschlossener Datenbanken mit Datenquellen von DB2 und Oracle

Die Umsatztabellen enthalten Spalten, in denen die Kundennummer (CUST_NO), die Bestellmenge (QUANTITY) und die Produktnummer (PROD_NO) aufgezeichnet werden. In der zusammengeschlossenen Datenbank ist darüber hinaus eine lokale Tabelle definiert, die Preisinformationen enthält. Diese Preistabelle umfasst Spalten, in denen die Produktnummer (PROD_NO) und der aktuelle Preis (PRICE) gespeichert werden.

Die Kurznamen für die fernen Datenquellenobjekte sind in den SYSCAT.TABLES-Tabellen gespeichert (vgl. hierzu die folgende Abbildung). In der Spalte TYPE ist der Typ des Objekts angegeben, wie Kurzname (K), lokale Tabelle (T) oder Sicht (S).

Datenquelleninformationen

Name des Datenquellenobjekts	Objekttyp	Speicherposition
PRICES	Lokale Tabelle	Zusammengeschlossene DB2-Datenbank
EUROPE_SALES	Ferne Tabelle	Datenbank von DB2 für z/OS und OS/390
US_SALES	Ferne Tabelle	Datenbank von DB2 für iSeries
JAPAN_SALES	Ferne Tabelle	Oracle-Datenbank
SALES_BY_REGION	Ferne Sicht	Oracle-Datenbank

SYSCAT.TABLES - Tabellen

TABELLENNAME	TYP
PRICES	T
FED_PRICES	K
Z_EU_SALES	K
iS_US_SALES	K
ORA_JAPANSALES	K
ORA_REGIONSALES	K
.....	

Abbildung 4. Tabellen und Kurznamen für Beispielabfragen

Im Folgenden sind Beispiele für die Anweisung SELECT in Verbindung mit dem Beispielsystem zusammenschlossener Datenbanken aufgeführt, das zuvor erläutert wurde.

Beispiel: Abfrage einer einzelnen Datenquelle:

Die Tabelle Z_EU_SALES enthält die von Ihren europäischen Kunden bestellten Produkte. Sie enthält auch die Bestellmenge für jeden Verkauf. Diese Abfrage verwendet eine Anweisung SELECT mit einer ORDER BY-Klausel, um die Umsatzzahlen für Europa aufzulisten und die Liste nach Kundennummern zu sortieren:

```
SELECT CUST_NO, PROD_NO, QUANTITY
   FROM Z_EU_SALES
   ORDER BY CUST_NO
```

Beispiel: Verknüpfung einer lokalen Datenquelle und einer fernen Datenquelle:

PRICES ist eine Tabelle, die sich in der zusammenschlossenen Datenbank befindet. Sie enthält die Preisliste für die von Ihnen verkauften Produkte. Sie möchten aus dieser Tabelle die Preise auswählen, die den in Z_EU_SALES aufgelisteten Produkten entsprechen. Außerdem möchten Sie die Ergebnismenge nach Kundennummern sortieren.

```
SELECT sales.CUST_NO, sales.PROD_NO, sales.QUANTITY
   FROM Z_EU_SALES sales, PRICES
   WHERE sales.PROD_NO=PRICES.PROD_NO
   ORDER BY sales.CUST_NO
```

Beispiel: Abfrage mehrerer ferner Datenquellen:

Angenommen, Sie möchten die Verkaufsdaten aller Regionen zusammenstellen und die Ergebnismenge nach Produktnummern sortieren.

```
WITH GLOBAL_SALES (Customer, Product, Quantity) AS
  (SELECT CUST_NO, PROD_NO, QUANTITY FROM Z_EU_SALES
   UNION ALL
   SELECT CUST.NO,PROD.NO, QUANTITY FROM iS_US_SALES
   UNION ALL
   SELECT CUST.NO,PROD.NO, QUANTITY FROM ORA_JAPANSALES)
SELECT Customer, Product, Quantity
   FROM GLOBAL_SALES
   ORDER BY Product
```

Angenommen, Sie haben in der Oracle-Datenquelle eine Sicht, die die Umsätze für Japan und Indonesien auflistet. Der Kurzname für diese Sicht ist ORA_SALESREGION. Sie möchten diese Informationen mit den Absätzen in den Vereinigten Staaten kombinieren und die Produktpreise und Absätze nebeneinander anzeigen.

```
SELECT us_jpn_ind.CUST_NO, us_jpn_ind.PROD_NO,
       us_jpn_ind.QUANTITY, us_jpn_ind.QUANTITY*PRICES.PRICE
       AS SALEPRICE FROM
       (SELECT CUST_NO, PROD_NO, QUANTITY
        FROM ORA_SALESREGION
 UNION ALL
 SELECT CUST_NO, PROD_NO, QUANTITY
 FROM iS_US_SALES us ) us_jpn_ind,PRICES
 WHERE us_jpn_ind.PROD_NO = PRICES.PROD_NO
 ORDER BY SALEPRICE DESC
```

Zugehörige Referenzen:

- „SELECT statement“ in *SQL Reference, Volume 2*

Ändern von Daten in einem System zusammengeschlossener Datenbanken

In einem System zusammengeschlossener Datenbanken können Sie Anweisungen des Typs INSERT, UPDATE und DELETE für Datenquellenobjekte unter Verwendung von Kurznamen absetzen. In den folgenden Abschnitten sind Beispiele für die Ausführung dieser Operationen enthalten.

Einfügen von Daten in Datenquellenobjekte

Um Daten in Datenquellen einzufügen, müssen Sie die Kurznamen der Datenquellenobjekte in der Anweisung INSERT verwenden.

Voraussetzungen:

Wenn Sie Daten unter Verwendung eines Kurznamens einfügen wollen, müssen alle folgenden Bedingungen für Zugriffsrechte erfüllt sein:

- Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen das Zugriffsrecht INSERT für den Kurznamen umfassen, damit die zusammengeschlossene Datenbank die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss über das Zugriffsrecht INSERT für das zugrunde liegende Tabellenobjekt verfügen, damit die Datenquelle die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss mit einer Benutzerzuordnung der Berechtigungs-ID auf dem Server mit zusammengeschlossenen Datenbanken zugeordnet werden.

Einschränkungen:

In zusammengeschlossenen Datenbanken werden INSERT-Operationen für einige Datenquellen nicht unterstützt. Informationen hierzu finden Sie im Abschnitt "Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammengeschlossener Datenbanken".

Vorgehensweise:

Angenommen, eine Informix-Tabelle besteht aus zwei Spalten. Die erste Spalte enthält hierbei INTEGER-Daten und die zweite Spalte VARCHAR-Daten (bis zu 20 Zeichen). Der Kurzname *infx_table_nn* ist auf dem Server mit zusammengeschlossenen Datenbanken für die Informix-Tabelle registriert.

Sie können für die Informix-Tabelle die Anweisungen INSERT, UPDATE und DELETE ausgeben und hierbei den Kurznamen *infx_table_nn* verwenden. Die Anweisung, die Sie angeben müssen, um z. B. eine neue Zeile mit Informationen in die Informix-Tabelle einzufügen, lautet wie folgt:

```
INSERT INTO db2user1.infx_table_nn VALUES(1,'Walter')
```

Zugehörige Konzepte:

- „Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammengeschlossener Datenbanken“ auf Seite 110

Zugehörige Tasks:

- „Auswählen von Daten in einem System zusammengeschlossener Datenbanken“ auf Seite 126
- „Aktualisieren von Daten in Datenquellenobjekten“ auf Seite 131
- „Löschen von Daten aus Datenquellenobjekten“ auf Seite 132

Zugehörige Referenzen:

- „INSERT statement“ in *SQL Reference, Volume 2*

Aktualisieren von Daten in Datenquellenobjekten

Um Daten in Datenquellen zu aktualisieren, müssen Sie in der Anweisung UPDATE die Kurznamen der Datenquellenobjekte verwenden.

Voraussetzungen:

Wenn Sie Daten unter Verwendung eines Kurznamens aktualisieren wollen, müssen alle folgenden Bedingungen für Zugriffsrechte erfüllt sein:

- Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen das Zugriffsrecht UPDATE für den Kurznamen umfassen, damit die zusammengeschlossene Datenbank die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss über das Zugriffsrecht UPDATE für das zugrunde liegende Tabellenobjekt verfügen, damit die Datenquelle die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss mit einer Benutzerzuordnung der Berechtigungs-ID auf dem Server mit zusammengeschlossenen Datenbanken zugeordnet werden.

Einschränkungen:

In zusammengeschlossenen Datenbanken werden UPDATE-Operationen für einige Datenquellen nicht unterstützt. Informationen hierzu finden Sie im Abschnitt "Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammengeschlossener Datenbanken".

Vorgehensweise:

Angenommen, eine Informix-Tabelle besteht aus zwei Spalten. Die erste Spalte enthält hierbei INTEGER-Daten und die zweite Spalte VARCHAR-Daten (bis zu 20 Zeichen). Der Kurzname *infx_table_nn* ist auf dem Server mit zusammengeschlossenen Datenbanken für die Informix-Tabelle registriert.

Sie können für die Informix-Tabelle die Anweisungen INSERT, UPDATE und DELETE ausgeben und hierbei den Kurznamen *infx_table_nn* verwenden. Die Anweisung, die Sie angeben müssen, um z. B. eine Zeile mit Informationen in der Informix-Tabelle zu aktualisieren, lautet wie folgt:

```
UPDATE db2user1.infx_table_nn SET c2='Bill' WHERE c1=2
```

Zugehörige Konzepte:

- „Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammengeschlossener Datenbanken“ auf Seite 110

Zugehörige Tasks:

- „Auswählen von Daten in einem System zusammengeschlossener Datenbanken“ auf Seite 126
- „Einfügen von Daten in Datenquellenobjekte“ auf Seite 130
- „Löschen von Daten aus Datenquellenobjekten“ auf Seite 132

Zugehörige Referenzen:

- „UPDATE statement“ in *SQL Reference, Volume 2*

Löschen von Daten aus Datenquellenobjekten

Um Daten aus Datenquellen zu löschen, müssen Sie die Kurznamen der Datenquellenobjekte in der Anweisung DELETE verwenden.

Voraussetzungen:

Wenn Sie Daten unter Verwendung eines Kurznamens löschen wollen, müssen alle folgenden Bedingungen für Zugriffsrechte erfüllt sein:

- Die Zugriffsrechte der Berechtigungs-ID der Anweisung müssen das Zugriffsrecht DELETE für den Kurznamen umfassen, damit die zusammengeschlossene Datenbank die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss über das Zugriffsrecht DELETE für das zugrunde liegende Tabellenobjekt verfügen, damit die Datenquelle die Anforderung akzeptiert.
- Die Benutzer-ID der Datenquelle muss mit einer Benutzerzuordnung der Berechtigungs-ID auf dem Server mit zusammengeschlossenen Datenbanken zugeordnet werden.

Einschränkungen:

In zusammengeschlossenen Datenbanken werden DELETE-Operationen für einige Datenquellen nicht unterstützt. Informationen hierzu finden Sie im Abschnitt "Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammengeschlossener Datenbanken".

Vorgehensweise:

Angenommen, eine Informix-Tabelle besteht aus zwei Spalten. Die erste Spalte enthält hierbei INTEGER-Daten und die zweite Spalte VARCHAR-Daten (bis zu 20 Zeichen). Der Kurzname *infx_table_nn* ist auf dem Server mit zusammengeschlossenen Datenbanken für die Informix-Tabelle registriert.

Sie können für die Informix-Tabelle die Anweisungen INSERT, UPDATE und DELETE ausgeben und hierbei den Kurznamen *infx_table_nn* verwenden. Die Anweisung, die Sie angeben müssen, um z. B. eine Zeile mit Informationen in der Informix-Tabelle zu löschen, lautet wie folgt:

```
DELETE FROM infx_table_nn WHERE c1=3
```

Zugehörige Konzepte:

- „Einschränkungen für die Anweisungen INSERT, UPDATE und DELETE in Systemen zusammengeschlossener Datenbanken“ auf Seite 110

Zugehörige Tasks:

- „Auswählen von Daten in einem System zusammengeschlossener Datenbanken“ auf Seite 126
- „Einfügen von Daten in Datenquellenobjekte“ auf Seite 130
- „Aktualisieren von Daten in Datenquellenobjekten“ auf Seite 131

Zugehörige Referenzen:

- „DELETE statement“ in *SQL Reference, Volume 2*

Zuordnungssemantik in einem System zusammengeschlossener Datenbanken

Wenn Sie einer Kurznamenspalte Daten zuordnen, kann der Datentyp auf der Basis der von DB2 Information Integrator verwendeten Zuordnungsregeln geändert werden. Um die erwarteten Ergebnisse zu erhalten, müssen Sie die Zuordnungsregeln kennen.

Im Folgenden finden Sie die Regeln zum Bestimmen des Zieldatentyps einer Zuordnung zu einer Kurznamenspalte:

- Festlegen des lokalen Quellentyps: Der lokale Quellentyp wird vom lokalen Spaltentyp und dem lokalen Ergebnistyp der Ausdrücke bestimmt. Wenn die Quelle konstant ist, stimmt der lokale Quellentyp mit dem Konstantentyp überein.
- Festlegen des Zieltyps:
 - Wenn die Zuordnungsquelle über keinen Typ verfügt (z. B. Parametermarken und Nullwerte (NULLs)), dann lautet der Zieltyp `MIN(local_target_type, remote_target_type)`. Dabei ist `local_target_type` der aktualisierte lokale Spaltendatentyp und `remote_target_type` ist der aktualisierte Spaltendatentyp der Datenquelle. `remote_target_type` verweist auf den progressiven Standardzuordnungstyp des Datentyps der fernen Zielspalte.
 - Wenn die Zuordnungsquelle ungleich NULL ist oder wenn es sich hierbei nicht um Parametermarken handelt, dann lautet der Zieltyp `MIN(local_target_type, remote_target_type, local_source_type)`.

Definition von MIN(type1, type2)

- Type1 und type2 sind nicht genau identisch.
- $\text{MIN}(\text{type1}, \text{type2}) = \text{MIN}(\text{type2}, \text{type1})$.
- $\text{MIN}(\text{type1}, \text{type2}) = \text{remote_target_type}(\text{local_target_type})$, wenn $\text{MIN}(\text{type1}, \text{type2}) = \text{DECIMAL}(0,0)$.
- BLOB ist nur mit BLOB kompatibel. Also gilt Folgendes: $\text{MIN}(\text{BLOB}(x), \text{BLOB}(y)) = \text{BLOB}(z)$, wobei $z = \min(x, y)$ ist.
- Die Datentypen TIME und DATE sind nicht kompatibel.
- Datums-/Zeitypen und Zeichenfolgen sind kompatibel.
- Bei Unicode-Datenbanken sind Zeichenfolgen und Grafikzeichenfolgen kompatibel.

In den folgenden Tabellen wird das Minimum von zwei Datentypen für numerische Datentypen, Zeichenfolge-, Grafikzeichenfolge- sowie für Datums- und Zeitdatentypen aufgelistet.

Tabelle 10. Numerische Datentypen

type1	type2	MIN(type1, type2)
SMALLINT	SMALLINT, INTEGER, BIGINT, REAL oder DOUBLE	SMALLINT
INTEGER	BIGINT, REAL oder DOUBLE	INTEGER
BIGINT	REAL oder DOUBLE	BIGINT
REAL	DOUBLE	REAL
DECIMAL(w,x)	SMALLINT	DECIMAL(p,0), wobei $p = w - x$ gilt, wenn $p < 5$; andernfalls SMALLINT
DECIMAL(w,x)	INTEGER	DECIMAL(p,0), wobei $p = w - x$ gilt, wenn $p < 11$; andernfalls INTEGER
DECIMAL(w,x)	BIGINT	DECIMAL(p,0), wobei $p = w - x$ gilt, wenn $p < 19$; andernfalls BIGINT
DECIMAL(w,x)	DECIMAL(y,z)	DECIMAL(p,s), wobei $p = \min(w, y) + \min(w - x, y - z)$ und $s = \min(x, z)$ gilt
DECIMAL(w,x)	DOUBLE oder REAL	DECIMAL(w,x)

In der folgenden Tabelle wird das Minimum von zwei Datentypen für Zeichenfolgedatentypen aufgelistet.

Tabelle 11. Zeichenfolgedatentypen

type1	type2	MIN(type1, type2)
CHAR(x)	CHAR(y), VARCHAR(y), LONG VARCHAR oder CLOB(y)	CHAR(z), wobei $z = \min(x, y)$ gilt
VARCHAR(x)	VARCHAR(y), LONG VARCHAR oder CLOB(y)	VARCHAR(z), wobei $z = \min(x, y)$ gilt
LONG VARCHAR	CLOB(y)	LONG VARCHAR, wobei $x > 32700$ und CLOB(x), wobei $x \leq 32700$ gilt
CLOB(x)	CLOB(y)	CLOB(z), wobei $z = \min(x, y)$ gilt

In der folgenden Tabelle wird das Minimum von zwei Datentypen für Grafikzeichenfolgedatentypen aufgelistet.

Tabelle 12. Grafikzeichenfolgedatentypen

type1	type2	MIN(type1, type2)
GRAPHIC(x)	GRAPHIC(y), VARGRAPHIC(y), LONG VARGRAPHIC oder DBCLOB(y)	GRAPHIC(z), wobei $z = \min(x, y)$ gilt
VARGRAPHIC(x)	VARGRAPHIC(y), LONG VARGRAPHIC oder DBCLOB(y)	VARGRAPHIC(z), wobei $z = \min(x, y)$ gilt
LONG VARGRAPHIC	DBCLOB(y)	LONG VARGRAPHIC, wobei $x > 32700$ und DBCLOB(x), wobei $x \leq 32700$ gilt
DBCLOB(x)	DBCLOB(y)	DBCLOB(z), wobei $z = \min(x, y)$ gilt

In der folgenden Tabelle wird das Minimum von zwei Datentypen für Datums- und Zeitdatentypen aufgelistet.

Tabelle 13. Datums- und Zeitdatentypen

type1	type2	MIN(type1, type2)
DATE	TIMESTAMP	DATE
TIME	TIMESTAMP	TIME

Wenn die von Ihnen eingefügten Daten vom Datentyp CHAR kürzer sind als die für das Ziel zulässige Länge, füllt die Datenquelle den Rest der Spalte mit Füllzeichen auf.

Wenn Sie Daten vom Typ DATE oder TIME in eine ferne Spalte vom Datentyp TIMESTAMP einfügen, füllt die Datenquelle den Rest der Spalte mit Füllzeichen auf.

Zuordnungssemantik in einem System zusammengesetzter Datenbanken - Beispiele

In Tabelle 14 finden Sie verschiedene Beispiele für die Anwendung einer zusammengesetzten Zuordnungssemantik in Abfragen mit einem lokalen und einem fernen Typ.

Tabelle 14. Beispiele für Zuordnungssemantik

Lokaler Typ	Ferner Typ	Ihre Abfrage	Generierte ferne Abfrage
FLOAT	INTEGER	Definieren Sie c1=123.23	Definieren Sie c1=INTEGER(123.23)
INTEGER	FLOAT	Definieren Sie c1=123.23	Definieren Sie c1=INTEGER(123.23)
FLOAT	INTEGER	Definieren Sie c1=123	Definieren Sie c1=123
CHAR(10)	CHAR(20)	Definieren Sie c1='123'	Definieren Sie c1='123' ('123' verfügt über den Typ VARCHAR(3); dabei handelt es sich um den kürzesten verfügbaren Datentyp)
CHAR(10)	CHAR(20)	Definieren Sie c1=char23col	Definieren Sie c1=CHAR(char23col, 10)
CHAR(10)	CHAR(20)	Definieren Sie c1=expr1	<ul style="list-style-type: none"> • Definieren Sie c1=expr1, wenn expr1 char(n) n<= 10 zurückgibt • Definieren Sie c1=CHAR(expr1, 10), wenn expr1 char(n) n>10 zurückgibt
TIMESTAMP	DATE	Definieren Sie c1 = aktuelle Zeitmarke	Definieren Sie c1=DATE (aktuelle Zeitmarke)

Kapitel 9. Überwachen eines Systems zusammenschlossener Datenbanken

In diesem Kapitel wird beschrieben, wie der Status von Kurznamen und Servern überwacht wird und wie mit dem Snapshot Monitor Abfragefragmente überwacht werden können.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Diagnoseanzeiger für Kurznamen und Server zusammenschlossener Datenbanken“
- „Aktivieren der Diagnoseanzeiger von Systemen zusammenschlossener Datenbanken“ auf Seite 139
- „Überwachen des Status von Kurznamen und Servern für zusammenschlossene Datenbanken“ auf Seite 139
- „Überwachen des Status von Kurznamen und Servern für zusammenschlossene Datenbanken - Beispiel“ auf Seite 140
- „Ausführen des Snapshot Monitor für Systeme zusammenschlossener Datenbanken - Übersicht“ auf Seite 141
- „Überwachen von Abfragefragmenten für zusammenschlossene Datenbanken“ auf Seite 142
- „Ausführen des Snapshot Monitor für Abfragefragmente für zusammenschlossene Datenbanken - Beispiel“ auf Seite 142

Diagnoseanzeiger für Kurznamen und Server zusammenschlossener Datenbanken

Sie können Diagnoseanzeiger in der DB2[®]-Diagnosezentrale verwenden, um den Status Ihrer Kurznamen und Server für Systeme zusammenschlossener Datenbanken zu überwachen. Der Diagnoseanzeiger für Kurznamen lautet `db.fed_nicknames_op_status`. Der Diagnoseanzeiger für Serverdefinitionen lautet `db.fed_servers_op_status`. Die Diagnoseanzeiger für Systeme zusammenschlossener Datenbanken werden bei der Installation des Diagnosemonitors installiert.

Standardmäßig aktiviert die Diagnosezentrale die Diagnoseanzeiger für Systeme zusammenschlossener Datenbanken nicht. Der Benutzer muss die Anzeiger aktivieren.

Wenn der Status eines Kurznamens oder Servers abnormal ist, geben die Diagnoseanzeiger eine Warnung aus. Sie können die Überwachungsergebnisse mit der Diagnosezentrale oder über die Befehlszeile anzeigen.

Diagnoseanzeiger werden von Servern mit zusammenschlossenen Datenbanken unterstützt, die die Betriebssysteme AIX[®], HP-UX, Linux, Microsoft[®] Windows[®] und Solaris verwenden.

In Tabelle 15 auf Seite 138 werden die Diagnoseanzeiger für Kurznamen und Server für Systeme zusammenschlossener Datenbanken beschrieben.

Tabelle 15. Diagnoseanzeiger für Kurznamen und Server

Diagnoseanzeiger	Beschreibung
db.fed_nicknames_op_status	<p>Gibt den Ergebnisstatus aller relationalen Kurznamen an, die in einer Datenbank auf einem DB2 UDB-Server mit zusammengeschlossenen Datenbanken definiert wurden.</p> <p>Gibt eine Warnung aus, wenn ein Kurzname ungültig ist. Enthält Details zu den ungültigen Kurznamen sowie Informationen zu Schritten, die Sie zur Behebung dieses Problems ausführen können.</p>
db.fed_servers_op_status	<p>Gibt den Ergebnisstatus aller Server mit zusammengeschlossenen Datenbanken an, die in einer Datenbank auf einem DB2 UDB-Server mit zusammengeschlossenen Datenbanken definiert wurden.</p> <p>Gibt eine Warnung aus, wenn ein Server nicht verfügbar ist. Enthält Details zu den nicht verfügbaren Servern sowie Informationen zu Schritten, die Sie zur Behebung dieses Problems ausführen können.</p>

Die Diagnoseanzeiger können die folgenden Datenquellen auswerten:

- DB2-Produktfamilie (DRDA)
- Excel
- Informix®
- Microsoft SQL Server
- ODBC
- Oracle (NET8)
- Sybase (CTLIB)
- Dateien mit Tabellenstruktur
- Teradata
- XML (nur Rootkurznamen)

Zugehörige Konzepte:

- „Introduction to the health monitor“ in *System Monitor Guide and Reference*

Zugehörige Tasks:

- „Überwachen des Status von Kurznamen und Servern für zusammengeschlossene Datenbanken“ auf Seite 139
- „Aktivieren der Diagnoseanzeiger von Systemen zusammengeschlossener Datenbanken“ auf Seite 139

Zugehörige Referenzen:

- „db2hc - Start Health Center Command“ in *Command Reference*
- „db.fed_nicknames_op_status - Nickname Status health indicator“ in *System Monitor Guide and Reference*
- „db.fed_servers_op_status - Data Source Server Status health indicator“ in *System Monitor Guide and Reference*

Aktivieren der Diagnoseanzeiger von Systemen zusammengesetzter Datenbanken

Sie müssen die Diagnoseanzeiger für Systeme zusammengesetzter Datenbanken aktivieren, um den Status von Kurznamen und Servern zu überwachen.

Der Diagnoseanzeiger für Kurznamen lautet `db.fed_nicknames_op_status`. Der Diagnoseanzeiger für Serverdefinitionen lautet `db.fed_servers_op_status`.

Vorgehensweise:

Öffnen Sie die DB2-Diagnosezentrale, und konfigurieren Sie die Diagnoseanzeiger, um die Diagnoseanzeiger für Systeme zusammengesetzter Datenbanken zu aktivieren.

Zugehörige Tasks:

- „Configuring health indicators using Health Center“ in *System Monitor Guide and Reference*
- „Überwachen des Status von Kurznamen und Servern für zusammengesetzte Datenbanken“ auf Seite 139

Überwachen des Status von Kurznamen und Servern für zusammengesetzte Datenbanken

Die Überwachung des Status von Kurznamen und Servern kann bei der Bestimmung und Lösung von Problemen mit Ihrem System zusammengesetzter Datenbanken nützlich sein. Mit Diagnoseanzeigern in der Diagnosezentrale können Sie den Status von Kurznamen und Servern für Systeme zusammengesetzter Datenbanken überwachen.

Sie können die Überwachungsergebnisse mit der Diagnosezentrale oder über die Befehlszeile anzeigen. Verwenden Sie die DB2-Steuerzentrale oder den DB2-Befehlszeilenprozessor, um die von den Diagnoseanzeigern festgestellten Fehler zu beheben.

Voraussetzungen:

- Stellen Sie sicher, dass die SELECT-Zugriffsrechte für die Kurznamen im DB2-Server mit zusammengesetzten Datenbanken definiert wurden.
- Setzen Sie den Konfigurationsparameter `FEDERATED` des Datenbankmanagers auf YES (JA).
- Wenn für die Datenquelle eine Authentifizierung erforderlich ist, muss die Datenquelle über Benutzerzuordnungen der ID des Diagnosemonitors verfügen. Der Diagnosemonitor verwendet diese Zuordnung zur Herstellung einer Verbindung zur Datenquelle.

Einschränkungen:

Die Diagnoseanzeiger können die folgenden Datenquellen nicht auswerten:

- BioRS
- BLAST
- Documentum
- Entrez

- Extended Search
- HMMER
- Web-Services
- WebSphere Business Integration
- XML (Nicht-Rootkurznamen)

Vorgehensweise:

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Öffnen Sie die Diagnosezentrale.
2. Öffnen Sie den Empfehlungsadvisor, um Empfehlungen zur Lösung der ungültigen Kurznamen oder der nicht verfügbaren Server anzuzeigen.

Geben Sie den Befehl GET HEALTH SNAPSHOT ein, um diese Task von der Befehlszeile aus auszuführen.

Zugehörige Konzepte:

- „Introduction to the health monitor“ in *System Monitor Guide and Reference*
- „Diagnoseanzeiger für Kurznamen und Server zusammenschlossener Datenbanken“ auf Seite 137

Zugehörige Tasks:

- „Resolving alerts using the Health Center“ in *System Monitor Guide and Reference*

Zugehörige Referenzen:

- „Überwachen des Status von Kurznamen und Servern für zusammenschlossene Datenbanken - Beispiel“ auf Seite 140

Überwachen des Status von Kurznamen und Servern für zusammenschlossene Datenbanken - Beispiel

Das folgende Beispiel zeigt die Diagnosemomentaufnahme einer Datenbank mit dem Namen fedhi. Die Namen der Diagnoseanzeiger für Systeme zusammenschlossener Datenbanken lauten db.fed_nicknames_op_status und db.fed_servers_op_status. In diesem Beispiel befinden sich beide Diagnoseanzeiger in einem normalen Status. 'Normal' bedeutet, dass die Kurznamen und Server gültig sind.

GET HEALTH SNAPSHOT FOR DATABASE ON fedhi:

Diagnosemomentaufnahme der Datenbank

Zeitmarke für Momentaufnahme	= 02/10/2004 12:10:55.063004
Datenbankname	= FEDHI
Datenbankpfad	= C:\DB2\NODE0000\SQL00006\
Aliasname der Eingabedatenbank	= FEDHI
Betriebssystem auf Datenbank-Server	= NT
Position der Datenbank	= Lokal
Höchster Alertstatus für Datenbank	= Achtung
Diagnoseanzeiger:	
Name des Diagnoseanzeigers	= db.fed_servers_op_status
Wert	= 0
Bewertungszeitmarke	= 02/10/2004 12:09:10.961000

Alertstatus	= Normal
Name des Diagnoseanzeigers	= db.fed_nicknames_op_status
Wert	= 0
Bewertungszeitmarke	= 02/10/2004 12:09:10.961000
Alertstatus	= Normal
Name des Diagnoseanzeigers	= db.db_op_status
Wert	= 0
Bewertungszeitmarke	= 02/10/2004 12:08:10.774000
Alertstatus	= Normal
Name des Diagnoseanzeigers	= db.sort_shrmem_util
Wert	= 0
Einheit	= %
Bewertungszeitmarke	= 02/10/2004 12:08:10.774000
Alertstatus	= Normal
Name des Diagnoseanzeigers	= db.spilled_sorts
Wert	= 0
Einheit	= %
Bewertungszeitmarke	= 02/10/2004 12:09:10.961000
Alertstatus	= Normal

Zugehörige Konzepte:

- „Diagnoseanzeiger für Kurznamen und Server zusammenschlossener Datenbanken“ auf Seite 137

Zugehörige Tasks:

- „Überwachen des Status von Kurznamen und Servern für zusammenschlossene Datenbanken“ auf Seite 139

Ausführen des Snapshot Monitor für Systeme zusammenschlossener Datenbanken - Übersicht

Sie können den Snapshot Monitor (Überwachungsprogramm für Momentaufnahmen) verwenden, um die zu einem bestimmten Zeitpunkt verfügbaren Informationen zu Datenquellen für Systeme zusammenschlossener Datenbanken und den zugehörigen Anwendungen zu erfassen. Momentaufnahmen dienen zur Ermittlung des Status eines Systems zusammenschlossener Datenbanken. Wenn Momentaufnahmen in regelmäßigen Abständen erstellt werden, können sie auch zur Beobachtung von Trends und zur Prognose potenzieller Fehler eingesetzt werden.

Zugehörige Konzepte:

- „Snapshot monitor“ in *System Monitor Guide and Reference*

Zugehörige Tasks:

- „Überwachen von Abfragefragmenten für zusammenschlossene Datenbanken“ auf Seite 142

Zugehörige Referenzen:

- Kapitel 31, „Systemmonitorelemente der zusammenschlossenen Datenbanken“, auf Seite 339
- „Ausführen des Snapshot Monitor für Abfragefragmente für zusammenschlossene Datenbanken - Beispiel“ auf Seite 142

Überwachen von Abfragefragmenten für zusammengeslossene Datenbanken

Durch die Überwachung von Abfragefragmenten haben Sie einen Überblick über die Leistung Ihres Systems zusammengeslossener Datenbanken. Wenn Sie Informationen zur Verarbeitung einer Abfrage durch ein System zusammengeslossener Datenbanken benötigen, können Sie eine Momentaufnahme der fernen Abfragesegmente abrufen.

Vorgehensweise:

Geben Sie den Befehl `GET SNAPSHOT FOR DYNAMIC SQL ON <dbname>` ein, um Abfragefragmente zu überwachen. Dabei ist *dbname* der Name der lokalen Datenbank auf dem Server mit zusammengeslossenen Datenbanken.

Zugehörige Konzepte:

- „Ausführen des Snapshot Monitor für Systeme zusammengeslossener Datenbanken - Übersicht“ auf Seite 141

Zugehörige Referenzen:

- „GET SNAPSHOT Command“ in *Command Reference*
- „Ausführen des Snapshot Monitor für Abfragefragmente für zusammengeslossene Datenbanken - Beispiel“ auf Seite 142

Ausführen des Snapshot Monitor für Abfragefragmente für zusammengeslossene Datenbanken - Beispiel

Im folgenden Beispiel sehen Sie die Ausgabe der Momentaufnahme für dynamische SQL, die für ein Abfragesegment erstellt wurde, das an eine ferne Oracle-Datenquelle mit dem Namen ORACLE817 gesendet wurde. Die Momentaufnahme wurde mit Hilfe der Anweisung `GET SNAPSHOT FOR DYNAMIC SQL ON LOCAL_FEDERATED_DATABASE` generiert. In den Ergebnissen sind Informationen für jede ferne Abfrage und für jede Abfrage im Cache für Anweisungen enthalten. Da Pufferpoolinformationen für ferne Abfragen nicht gelten, wurden in der Momentaufnahme keine Pufferpoolinformationen erfasst.

Dynamisches SQL - Momentaufnahmenergebnis

Anzahl Ausführungen	= 1
Anzahl Kompilierungen	= 1
Schlechteste Vorbereitungszeit (ms)	= 215
Beste Vorbereitungszeit (ms)	= 215
Intern gelöschte Zeilen	= 0
Intern eingefügte Zeilen	= 0
Gelesene Zeilen	= 25412
Intern aktualisierte Zeilen	= 0
Geschriebene Zeilen	= 25410
Sortiervorgänge für Anweisungen	= 0
Logische Lesevorgänge im Pufferpool	= Nicht gesammelt
Physische Lesevorgänge im Pufferpool	= Nicht gesammelt

```

|           Logische Lesevorgänge in temporären Daten des Pufferpools = Nicht gesammelt
|           Physische Lesevorgänge in temporären Daten des Pufferpools = Nicht gesammelt
|           Logische Lesevorgänge im Pufferpoolindex = Nicht gesammelt
|           Physische Lesevorgänge im Pufferpoolindex = Nicht gesammelt
|           Logische Lesevorgänge im temporären Pufferpoolindex = Nicht gesammelt
|           Physische Lesevorgänge im temporären Pufferpoolindex = Nicht gesammelt
|           Gesamte Ausführungszeit (sek.ms) = 20.229786
|           Gesamte Benutzer-CPU-Zeit (sek.ms) = 10.080000
|           Gesamte System-CPU-Zeit (sek.ms) = 0.520000
|           Anweisungstext = [ORACLE817]SELECT A0.C1,A0.C2 FROM ORA_T A0
|                               WHERE A0.C3 = :H0
|

```

Zugehörige Konzepte:

- „Ausführen des Snapshot Monitor für Systeme zusammengeschnittener Datenbanken - Übersicht“ auf Seite 141

Zugehörige Tasks:

- „Überwachen von Abfragefragmenten für zusammengeschnittene Datenbanken“ auf Seite 142

Zugehörige Referenzen:

- „GET SNAPSHOT Command“ in *Command Reference*

Kapitel 10. Unicode-Unterstützung für Datenquellen zusammengesetzter Datenbanken

In diesem Kapitel wird die Unicode-Unterstützung in einem System zusammengesetzter Datenbanken beschrieben.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Unicode-Unterstützung für Systeme zusammengesetzter Datenbanken“
- „Angabe der Client-Codepage für die Unicode-Unterstützung von Microsoft SQL Server- und ODBC-Datenquellen“ auf Seite 147
- „Unterstützte Unicode-Codepages für die Option CODEPAGE des MSSQL- und ODBC-Wrappers“ auf Seite 148
- „Angabe der Dateicodepage für die Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur“ auf Seite 149
- „Angabe der Dateicodepage für Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur - Beispiel“ auf Seite 149
- „Fehler auf Grund des Unterschieds zwischen der Größe des fernen Codepunkts und der Größe des Codepunkts für zusammengesetzte Datenbanken“ auf Seite 150

Unicode-Unterstützung für Systeme zusammengesetzter 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 zusammengesetzten 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[®]
 - Informix[®]
 - 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[®] Lotus[®] Extended Search-Wrapper
 - Benutzerdefinierte KEGG-Funktionen

- Benutzerdefinierte MQ-Funktionen
- Wrapper für Dateien mit Tabellenstruktur
- Benutzerdefinierte Web-Services-Funktionen
- Web-Services-Wrapper
- WebSphere® Business Integration-Wrapper
- XML-Wrapper

In Abb. 5 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® 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 zusammenschlossenen Datenbank Unicode verwenden. Anschließend kann das Unternehmen die Tabellen verknüpfen, um die Gesamtzahl aller Bestellungen unabhängig vom jeweiligen Gebiet anzuzeigen.

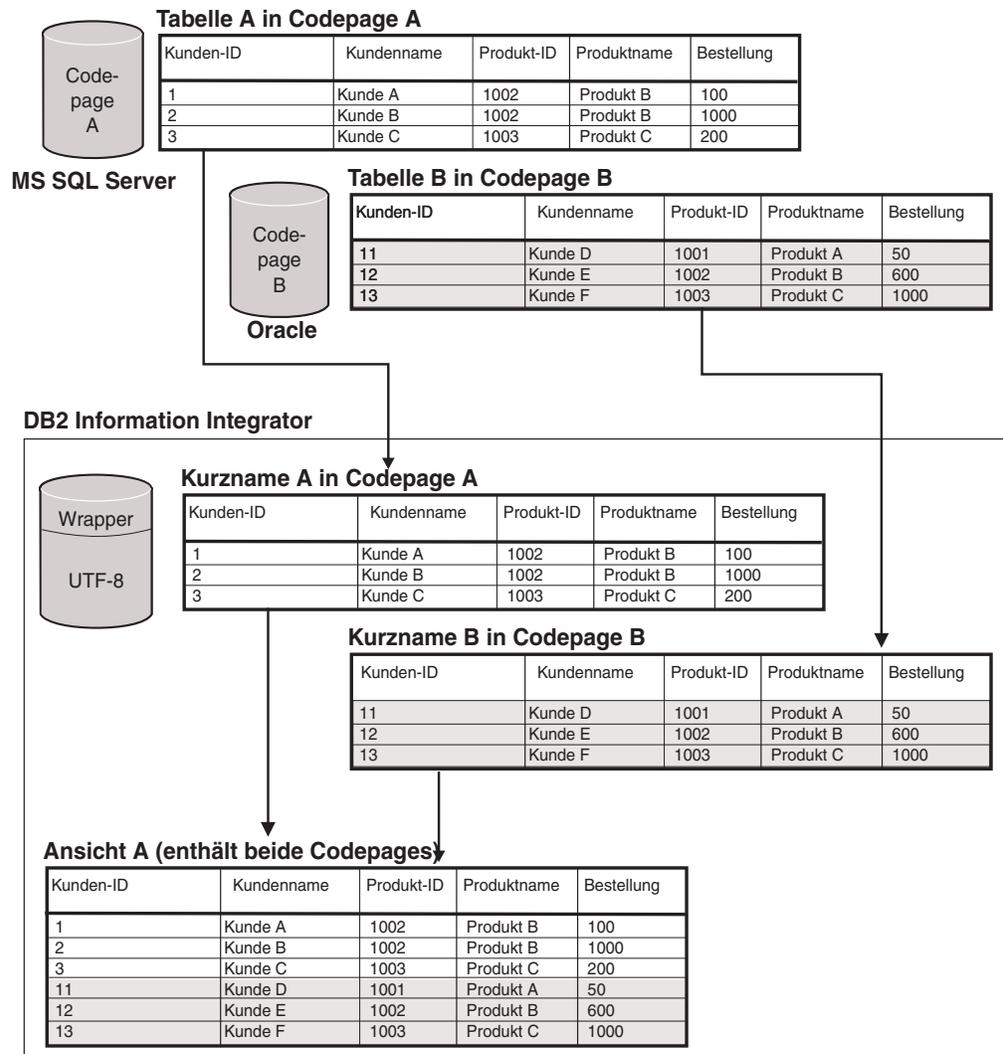


Abbildung 5. Unicode - Beispiel

Zugehörige Tasks:

- „Angabe der Client-Codepage für die Unicode-Unterstützung von Microsoft SQL Server- und ODBC-Datenquellen“ auf Seite 147
- „Angabe der Dateicodepage für die Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur“ auf Seite 149

Zugehörige Referenzen:

- „Standardwerte für progressive Unicode-Datentypzuordnungen - NET8-Wrapper“ auf Seite 327
- „Unterstützte Unicode-Codepages für die Option CODEPAGE des MSSQL- und ODBC-Wrappers“ auf Seite 148
- „Standardmäßige regressive Unicode-Datentypzuordnungen - NET8-Wrapper“ auf Seite 327
- „Standardmäßige progressive Unicode-Datentypzuordnungen - Sybase-Wrapper“ auf Seite 328
- „Standardmäßige regressive Unicode-Datentypzuordnungen - Sybase-Wrapper“ auf Seite 329
- „Standardwerte für progressive Unicode-Datentypzuordnungen - ODBC-Wrapper“ auf Seite 329
- „Standardmäßige regressive Unicode-Datentypzuordnungen - ODBC-Wrapper“ auf Seite 330
- „Standardmäßige progressive Unicode-Datentypzuordnungen - Microsoft SQL Server-Wrapper“ auf Seite 330
- „Standardmäßige regressive Unicode-Datentypzuordnungen - Microsoft SQL Server-Wrapper“ auf Seite 331
- „Angabe der Dateicodepage für Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur - Beispiel“ auf Seite 149

Angabe der Client-Codepage für die Unicode-Unterstützung von Microsoft SQL Server- und ODBC-Datenquellen

Um die richtige Codepagekonvertierung für Microsoft SQL Server- und ODBC-Datenquellen sicherzustellen, müssen Sie die Client-Codepage angeben, wenn sich die Codepage von der Codepage der zusammengeschlossenen Datenbank unterscheidet.

Vorgehensweise:

Geben Sie eine Anweisung CREATE SERVER ein, wobei die Option CODEPAGE den Wert der Client-Codepage hat.

Beispiel: Wenn für die Datenquelle Microsoft SQL Server verwendet wird, der Server mit zusammengeschlossenen Datenbanken unter Windows ausgeführt wird und die Standardländereinstellung des Betriebssystems auf Japanisch (Shift-JIS) gesetzt ist, muss die Serveroption CODEPAGE entweder auf 943 (Shift-JIS) oder 1202 (UTF-16LE) gesetzt werden.

Geben Sie die folgende Anweisung ein, um die Codepage 1202 für die Microsoft SQL Server-Datenquelle mit dem Namen FEDSERVERW anzugeben:

```
CREATE SERVER FEDSERVERW TYPE MSSQLSERVER VERSION 2000 WRAPPER MSSQLODBC3
    OPTIONS(NODE 'SAMPLE', DBNAME 'TESTDB', CODEPAGE '1202');
```

Wenn für die Datenquelle Microsoft SQL Server verwendet wird, der Server mit zusammengeschlossenen Datenbanken unter UNIX ausgeführt wird und die App-CodePage- bzw. die IANAAppCodePage-Einstellung des DataDirect Connect-Clients auf 6 (Shift-JIS) gesetzt ist, muss die Serveroption CODEPAGE entweder auf 943 (Shift-JIS) oder 1208 (UTF-8) gesetzt werden.

Geben Sie die folgende Anweisung ein, um die Codepage 1208 für die Microsoft SQL Server-Datenquelle mit dem Namen FEDSERVERU anzugeben:

```
CREATE SERVER FEDSERVERU TYPE MSSQLSERVER VERSION 2000 WRAPPER MSSQLODBC3
    OPTIONS(NODE 'SAMPLE', DBNAME 'TESTDB', CODEPAGE '1208');
```

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammengeschlossener Datenbanken“ auf Seite 145

Zugehörige Referenzen:

- „CREATE SERVER statement“ in *SQL Reference, Volume 2*
- Kapitel 21, „Serveroptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 253
- „Unterstützte Unicode-Codepages für die Option CODEPAGE des MSSQL- und ODBC-Wrappers“ auf Seite 148

Unterstützte Unicode-Codepages für die Option CODEPAGE des MSSQL- und ODBC-Wrappers

Gültige Codepagewerte sind Werte, die von DB2 Universal Database unterstützt werden. Weitere gültige Werte finden Sie in Tabelle 16.

Tabelle 16. Unterstützte Unicode-Codepages für die Option CODEPAGE des MSSQL- und des ODBC-Wrappers

Wert der Option CODEPAGE	Beschreibung
1200	Codepage1200 - UCS-2 (Big-Endian)
1202	Codepage1202 - UCS-2 (Little-Endian)
1208	Codepage1208 - UTF-8
1232	Codepage1232 - UTF-32 (Big-Endian)
1234	Codepage1234 - UTF-32 (Little-Endian)

Zugehörige Tasks:

- „Angeben der Client-Codepage für die Unicode-Unterstützung von Microsoft SQL Server- und ODBC-Datenquellen“ auf Seite 147

Zugehörige Referenzen:

- „Unterstützte Gebietscodes und Codepages“ in *Systemverwaltung: Konzept*

Angeben der Dateicodpage für die Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur

Um die richtige Codpagekonvertierung für Datenquellen in Dateien mit Tabellenstruktur sicherzustellen, müssen Sie die Dateicodpage angeben, wenn sich die Codpage von der Codpage der zusammengeschlossenen Datenbank unterscheidet.

Einschränkungen:

Sie können die Option CODEPAGE nur in einer zusammengeschlossenen Unicode-Datenbank verwenden.

Vorgehensweise:

Geben Sie die Anweisung CREATE NICKNAME ein, wobei die Option CODEPAGE den Wert der Codpage der Daten in der Datei mit Tabellenstruktur hat. Gültig sind die Werte, die von DB2 Universal Database unterstützt werden. Der Standardwert ist die Codpage der zusammengeschlossenen DB2 Universal Database-Datenbank.

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammengeschlossener Datenbanken“ auf Seite 145

Zugehörige Referenzen:

- „CREATE NICKNAME statement“ in *SQL Reference, Volume 2*
- „Unterstützte Gebietscodes und Codpages“ in *Systemverwaltung: Konzept*
- „Angeben der Dateicodpage für Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur - Beispiel“ auf Seite 149
- „CREATE NICKNAME-Anweisungssyntax - Wrapper für Dateien mit Tabellenstruktur“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Angeben der Dateicodpage für Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur - Beispiel

Die Codpage der Daten in einer Datei mit dem Namen DRUGDATA1.TXT lautet 943. Geben Sie die folgende Anweisung CREATE NICKNAME ein, um als Codpage einer Datei mit Tabellenstruktur 943 anzugeben:

```
CREATE NICKNAME DRUGDATA1(Dcode Integer NOT NULL, Drug CHAR(20),
    Manufacutuer CHAR(20))
FOR SERVER biochem_lab
    OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT',CODEPAGE '943',
    COLUMN_DELIMITER '.',
    SORTED 'Y', KEY_COLUMN 'DCODE', VALIDATE_DATA_FILE 'Y');
```

Zugehörige Tasks:

- „Angeben der Dateicodpage für die Unicode-Unterstützung für Datenquellen in Dateien mit Tabellenstruktur“ auf Seite 149

Fehler auf Grund des Unterschieds zwischen der Größe des fernen Codepunkts und der Größe des Codepunkts für zusammengeslossene Datenbanken

Wenn sich die Größe des Codepunkts zwischen der zusammengeslossenen Datenbank und der fernen Datenquelle unterscheidet, werden möglicherweise falsche Daten zurückgegeben oder der Einfügevorgang schlägt fehl.

Wenn die zusammengeslossene Datenbank über eine höhere Codepunktgröße als die ferne Datenquelle verfügt, kann der Server mit zusammengeslossenen Datenbanken die Daten abschneiden, die Sie aus der fernen Datenquelle auswählen. Daten werden abgeschnitten, wenn bei der Konvertierung von Zeichenfolgen eine höhere Anzahl von Byte generiert wird, als für die Größe der Kurznamenspalte definiert ist. Die verbleibenden Byte werden nicht belegt. Sie können auch Daten einfügen, die größer als die Kurznamenspalte sind. Dieser Einfügungstyp kann erfolgreich verwendet werden, wenn bei der Konvertierung eine Anzahl von Byte generiert wird, die kleiner oder gleich dem Wert für die Größe der fernen Spalte ist.

Wenn die zusammengeslossene Datenbank über eine niedrigere Codepunktgröße verfügt als die ferne Datenquelle, kann es zu einem Fehlschlagen des Einfügevorgangs für Daten kommen. Einfügevorgänge schlagen fehl, wenn bei der Konvertierung von Zeichenfolgen eine höhere Anzahl von Byte generiert wird, als für die Größe der fernen Datenquellenspalte definiert ist.

Stellen Sie sicher, dass die Größen der Codepunkte zwischen der zusammengeslossenen Datenbank und der fernen Datenquelle sich nicht in einem Maß unterscheiden, das zu den oben erläuterten Fehlern führt.

Teil 3. Leistung

Kapitel 11. Optimieren der Leistung eines Systems zusammengesessener Datenbanken

Leistungsprobleme können ihren Ursprung in der zusammengesessenen Datenbank und/oder in den Datenquellen haben. Ein in der zusammengesessenen Datenbank oder in den Datenquellen entstehender Engpass kann die Leistung mindern. Die Problemerkennung umfasst die Optimierung der zusammengesessenen Datenbank und der Datenquellen im Hinblick auf eine maximale Leistung. Eventuell ist die Optimierung von Abfragen, Anwendungen, Konfigurationsparametern und Netzwerkbenutzung für die Problemlösung erforderlich.

In diesem Kapitel wird Folgendes behandelt:

- „Veröffentlichungen zur Leistung zusammengesessener Datenbanken“
- „Optimieren der Abfrageverarbeitung“
- „Pushdown-Analyse“ auf Seite 156
- „Globale Optimierung“ auf Seite 170
- „Systemmonitorelemente, die die Leistung beeinflussen“ auf Seite 179

Veröffentlichungen zur Leistung zusammengesessener Datenbanken

Die folgenden IBM Dokumente enthalten ausführliche Informationen zur Leistungsoptimierung:

- *Data Federation with IBM DB2 Information Integrator V8.1*, unter der Webadresse [http://publib-boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg247052.html?Open](http://publib.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg247052.html?Open)
- "Using the federated database technology of IBM DB2 Information Integrator", unter der Webadresse <ftp://ftp.software.ibm.com/software/data/pubs/papers/iifed.pdf>
- "DB2 Information Integrator XML Wrapper Performance", unter der Webadresse <ftp://ftp.software.ibm.com/software/data/pubs/papers/db2iixmlwrapper.pdf>

Optimieren der Abfrageverarbeitung

Zum Abrufen von Daten aus Datenquellen übergeben Clients (Benutzer und Anwendungen) Abfragen in DB2[®]-SQL an die zusammengesessene Datenbank. Zum Verarbeiten der Abfrage zieht der DB2-SQL-Compiler Informationen aus dem globalen Katalog und dem Wrapper der Datenquelle hinzu. Dabei handelt es sich um Informationen zur Verbindung zur Datenquelle, zu Serverattributen, Zuordnungen, Indexinformationen und Kurznamenstatistiken.

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 zusammengesessenen Datenbanken erfolgen.

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

DB2 UDB wertet die Zugriffspläne in erster Linie auf der Grundlage von Informationen über das Leistungsspektrum der Datenquelle und über die 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. Dazu gehören:

- Die zu verarbeitende Datenmenge
- Die Verarbeitungsgeschwindigkeit der Datenquelle
- Die vom Fragment zurückgegebene Datenmenge
- Die Bandbreite der Übertragung

| Die Pushdown-Analyse wird nur für relationale Datenquellen ausgeführt. Nicht
| relationale Datenquellen verwenden das RRC-Protokoll (RRC = Request-Reply-
| Compensate).

In der folgenden Abbildung werden die vom SQL-Compiler ausgeführten Schritte bei der Verarbeitung einer Abfrage gezeigt.

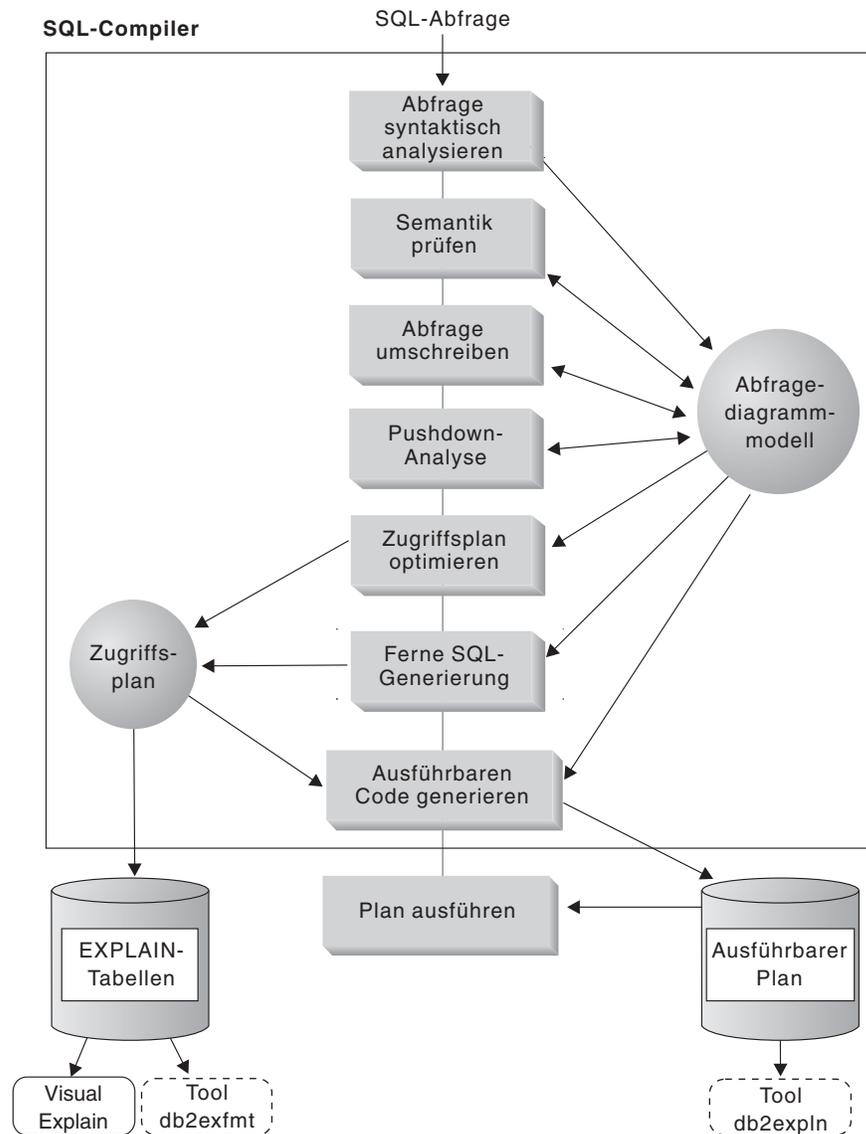


Abbildung 6. Flussdiagramm einer Abfrageanalyse des SQL-Compilers

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

Sollen Datenquellen Fragmente verarbeiten, 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.

Die Pushdown-Analyse soll im Wesentlichen feststellen, welche Operationen fern ausgewertet werden können. Die Pushdown-Analyse stellt dies auf der Basis der empfangenen SQL-Anweisung und der Funktionalität der fernen Datenquelle fest. Anhand dieser Analyse wertet das Abfrageoptimierungsprogramm die Alternativen aus und wählt auf der Basis des anfallenden Aufwands einen Zugriffsplan aus.

Das Optimierungsprogramm führt unter Umständen eine Verarbeitung aus Gründen der Aufwandseffizienz nicht direkt in der fernen Datenquelle aus. Ferner hat die Pushdown-Analyse die Aufgabe, die Abfrage möglichst in ein Format umzusetzen, das vom DB2-Optimierungsprogramm und von fernen Abfrageoptimierungsprogrammen besser optimiert werden kann.

Der vom Optimierungsprogramm ausgewählte endgültige Zugriffsplan kann aus Verarbeitungen in der fernen Datenquelle bestehen. Für Verarbeitungen, die von der jeweiligen Datenquelle ausgeführt werden, erstellt die ferne SQL-Generierung eine auf der SQL-Programmversion der Datenquelle basierte effiziente SQL-Anweisung. Dieser Vorgang unterstützt das Erstellen eines optimalen Zugriffsplans für die Abfrage aller Datenquellen und wird als *globale Optimierung* bezeichnet.

Für nicht relationale Quellen verwenden die Wrapper das RRC-Protokoll.

Zugehörige Konzepte:

- „Pushdown-Analyse“ auf Seite 156

Zugehörige Tasks:

- „Request-reply-compensate protocol“ in *IBM DB2 Information Integrator Wrapper Developer's Guide*
- „Globale Optimierung“ auf Seite 170

Pushdown-Analyse

Die Pushdown-Analyse wird für relationale Datenquellen ausgeführt. Nicht relationale Datenquellen verwenden das RRC-Protokoll. Sie gibt dem Abfrageoptimierungsprogramm an, ob eine ferne Datenquelle eine Operation ausführen kann. Eine *Operation* kann eine Funktion sein, wie z. B. relationale Bediener-, System- oder Benutzerfunktionen, oder ein SQL-Operator (GROUP BY, ORDER BY, usw.).

Funktionen, die nicht im Pushdown-Modus weitergeleitet werden können, wirken sich unter Umständen erheblich auf die Abfrageleistung aus. Beachten Sie die Auswirkungen, wenn ein selektives Vergleichselement erzwungen lokal ausgewertet wird anstatt in der fernen Datenquelle. Bei dieser Methode ist es möglicherweise erforderlich, dass der Server mit zusammengeschlossenen Datenbanken die gesamte Tabelle von der fernen Datenquelle abrufen und die Tabelle dann lokal mit dem Vergleichselement filtern muss. Steht das Netzwerk nur beschränkt zur Verfügung und ist die Tabelle sehr groß, kann dies die Abfrageleistung möglicherweise mindern.

Nicht im Pushdown-Modus weitergeleitete Operatoren können sich ebenfalls erheblich auf die Abfrageleistung auswirken. Das lokale Zusammenfassen von fernen Daten durch einen Operator GROUP BY beispielsweise könnte ebenso erfordern, dass der Server mit zusammengeschlossenen Datenbanken die gesamte Tabelle von der fernen Datenquelle abrufen muss.

Beispiel: Der Kurzname EMP verweist auf die Tabelle EMPLOYEE. Diese Tabelle hat 10.000 Zeilen. Eine Spalte enthält die Nachnamen der Mitarbeiter, eine andere Spalte enthält die Gehälter der Mitarbeiter. Die folgende Abfrage wird an den Server mit zusammengeschlossenen Datenbanken gesendet, um die Anzahl der Mitarbeiter zu berechnen, deren Nachname mit 'B' beginnt und deren Gehalt höher als 50.000 ist.

```

SELECT LASTNAME, COUNT (*) FROM EMP
WHERE LASTNAME LIKE 'B%' AND SALARY > '50000'
GROUP BY LASTNAME;

```

Empfängt der DB2®-SQL-Compiler diese Anweisung, überprüft er mehrere Möglichkeiten:

- Die Sortierfolgen sind gleich. Es ist wahrscheinlich, dass das Abfragevergleichselement im Pushdown-Modus an die Datenquelle weitergeleitet wird. Normalerweise ist es effizienter, Ergebnisse in der Datenquelle zu filtern und zu gruppieren anstatt die gesamte Tabelle auf den Server mit zusammengeschlossenen Datenbanken zu kopieren und die Operationen lokal auszuführen. Die Pushdown-Analyse ermittelt, ob Operationen in der Datenquelle ausgeführt werden können. Da die Sortierfolgen gleich sind, können die Operationen für das Vergleichselement und für GROUP BY in der Datenquelle ausgeführt werden.
- Die Sortierfolgen sind gleich, und das Abfrageoptimierungsprogramm kalkuliert die hohe Verarbeitungsgeschwindigkeit des Servers mit zusammengeschlossenen Datenbanken mit ein. Möglicherweise ermittelt das Abfrageoptimierungsprogramm die lokale Ausführung der Operation GROUP BY als die beste Methode (d. h. als die Methode mit dem geringsten anfallenden Aufwand). Das Vergleichselement wird zur Auswertung im Pushdown-Modus an die Datenquelle weitergeleitet. Hierbei handelt es sich um ein Beispiel für eine Kombination aus Pushdown-Analyse und globaler Optimierung.
- Die Sortierfolgen sind nicht gleich. Die Pushdown-Analyse ermittelt, dass die gesamte Klausel WHERE nicht in der Datenquelle ausgewertet werden kann. Das Abfrageoptimierungsprogramm ermittelt jedoch möglicherweise, dass der Abschnitt LIKE des Vergleichselements aus Gründen der Effizienz ausgelagert werden soll. Der Bereichsvergleich muss in der zusammengeschlossenen Datenbank durchgeführt werden. Hierbei handelt es sich um ein weiteres Beispiel für eine Kombination aus Pushdown-Analyse und globaler Optimierung.

Der SQL-Compiler überprüft die verfügbaren Zugriffspläne und wählt anschließend den effizientesten aus.

Grundsätzlich soll sichergestellt werden, dass das Abfrageoptimierungsprogramm das Weiterleiten von Funktionen und Operatoren zur Auswertung an die Datenquellen in Betracht zieht. Viele Faktoren bestimmen, ob eine Funktion oder ein SQL-Operator in einer fernen Datenquelle ausgewertet wird. Das Abfrageoptimierungsprogramm wird von folgenden Hauptfaktoren beeinflusst: Kenndaten des Servers, Kenndaten des Kurznamens und Kenndaten der Abfrage.

Zugehörige Konzepte:

- „Serverkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 158
- „Kurznamenkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 163
- „Abfragekenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 165

Zugehörige Tasks:

- „Request-reply-compensate protocol“ in *IBM DB2 Information Integrator Wrapper Developer's Guide*

Pushdown-Analyse - ausführliche Informationen

Serverkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten

In den folgenden Abschnitten werden datenquellenspezifische Faktoren beschrieben, die Pushdown-Möglichkeiten beeinflussen können.

Diese Faktoren treten bei relationalen Datenquellen auf, da zur Übergabe von Abfragen die SQL-Programmversion von DB2[®] verwendet wird und diese unter Umständen eine höhere Funktionalität bietet als die SQL-Programmversion der Datenquelle. Der DB2-Server mit zusammengeschlossenen Datenbanken kann die niedrigere Funktionalität eines Datenservers ausgleichen. Dafür kann es jedoch erforderlich sein, dass die Operation auf dem Server mit zusammengeschlossenen Datenbanken ausgeführt wird.

Die Pushdown-Möglichkeiten werden bei nicht relationalen Datenquellen von anderen Faktoren beeinflusst als bei relationalen Datenquellen. Die SQL-Programmversion spielt für die meisten nicht relationalen Datenquellen beispielsweise keine Rolle, da sie kein SQL verwenden.

Unterschiede bei SQL

- Leistungsspektrum von SQL. Jede Datenquelle unterstützt eine Variante der SQL-Programmversion und verschiedene Funktionalitätsebenen. Ein Beispiel ist die Liste GROUP BY. Die meisten Datenquellen unterstützen den Operator GROUP BY. Bei manchen Datenquellen bestehen jedoch Einschränkungen für die Anzahl der Elemente oder für die Zulassung eines Ausdrucks auf der Liste GROUP BY. Besteht in der fernen Datenquelle eine Einschränkung, führt der Server mit zusammengeschlossenen Datenbanken die Operation GROUP BY möglicherweise lokal aus.
- Einschränkungen für SQL. Datenquellen können unterschiedliche Einschränkungen für SQL haben. Einige Datenquellen erfordern z. B. Parametermarken für die Einbindung von Werten für ferne SQL-Anweisungen. Einschränkungen für Parametermarken müssen daher überprüft werden, um sicherzustellen, dass alle Datenquellen einen solchen Bindemechanismus unterstützen. Kann der Server mit zusammengeschlossenen Datenbanken keine passende Methode für die Einbindung eines Wertes in eine Funktion ermitteln, muss diese Funktion lokal ausgewertet werden.
- Begrenzungen für SQL. Der Server mit zusammengeschlossenen Datenbanken lässt möglicherweise die Verwendung von größeren ganzen Zahlen zu als die fernen Datenquellen. Werte, die die Begrenzung überschreiten, können jedoch nicht in Anweisungen eingebettet werden, die an die Datenquellen gesendet werden. Die Funktion oder der Operator, der mit dieser Konstante operiert, muss daher lokal ausgewertet werden.
- Serverspezifikationen. Zu dieser Kategorie gehören mehrere Faktoren. Ein Beispiel ist die Sortierung von NULL-Werten (je nach Sortierfolge höchster oder niedrigster Wert). Wird ein NULL-Wert beispielsweise in der Datenquelle nach anderen Kriterien sortiert als auf dem Server mit zusammengeschlossenen Datenbanken, können Operationen ORDER BY für einen Ausdruck mit Nullwerten nicht fern ausgewertet werden.

Sortierfolge

Wenn Sie die Serveroption `COLLATING_SEQUENCE` auf 'Y' setzen, teilen Sie der zusammengeschlossenen Datenbank mit, dass die Sortierfolge der Datenquelle mit der Sortierfolge von DB2 übereinstimmt. Durch diese Einstellung kann das Optimierungsprogramm sortierungsabhängige Verarbeitungen in einer Datenquelle berücksichtigen, wodurch eine bessere Leistung erzielt werden kann.

Weicht die Sortierfolge der Datenquelle von der Sortierfolge der zusammengeschlossenen Datenbank ab, erhalten Sie unter Umständen fehlerhafte Ergebnisse. Verwendet der Zugriffsplan beispielsweise Mischverknüpfungen, leitet das Optimierungsprogramm so viele Sortieroperationen wie möglich im Pushdown-Modus an die Datenquellen weiter. Verfügt die Datenquelle nicht über dieselbe Sortierfolge, ist die Ergebnismenge der Verknüpfungsergebnisse möglicherweise falsch. Setzen Sie die Serveroption `COLLATING_SEQUENCE` auf 'N', wenn Unklarheit darüber besteht, ob die Sortierfolge der Datenquelle mit der Sortierfolge von DB2 übereinstimmt.

Sie können jedoch auch eine zusammengeschlossene Datenbank so konfigurieren, dass sie die gleiche Sortierfolge verwendet wie die betreffende Datenquelle. Setzen Sie dann die Serveroption `COLLATING_SEQUENCE` auf 'Y'. Dadurch kann das Optimierungsprogramm die Durchführung eines Pushdown-Prozesses für Vergleichselemente von Zeichenbereichen in Betracht ziehen.

Überprüfen Sie die folgenden Faktoren, um festzustellen, ob die Datenquelle und DB2 UDB über die gleiche Sortierfolge verfügen:

- Unterstützung in der Landessprache
Die Sortierfolge bezieht sich auf die von einem Server unterstützte Sprache. Vergleichen Sie die Informationen über die Unterstützung in der Landessprache von DB2 für Ihr Betriebssystem mit den entsprechenden Informationen, die für die Datenquelle gelten.
- Merkmale der Datenquelle
Einige Datenquellen werden unter Verwendung von Sortierfolgen erstellt, die von der Groß-/Kleinschreibung unabhängig sind, was in DB2 UDB bei sortierungsabhängigen Operationen unterschiedliche Ergebnisse zur Folge haben kann.
- Anpassung
Einige Datenquellen bieten mehrere Optionen für Sortierfolgen oder erlauben eine Anpassung der Sortierfolge.

Erfordert eine Abfrage von einem Server mit zusammengeschlossenen Datenbanken eine Sortierung, hängt der Ort für die Durchführung der Sortierung von mehreren Faktoren ab. Sind die Sortierfolgen in der zusammengeschlossenen Datenbank und in der Datenquelle gleich, kann die Sortierung in der Datenquelle ausgeführt werden. Sind die Sortierfolgen gleich, kann das Abfrageoptimierungsprogramm bestimmen, ob eine lokale Sortierung oder eine ferne Sortierung die effizientere Lösung ist. Ebenso kann ein durch eine Abfrage eventuell erforderlicher Vergleich der Zeichendaten auch in der Datenquelle ausgeführt werden.

Numerische Vergleiche können im Allgemeinen selbst bei unterschiedlicher Sortierfolge an beiden Orten durchgeführt werden. Es können jedoch fehlerhafte Ergebnisse auftreten, wenn die Gewichtung von Nullzeichen in der zusammengeschlossenen Datenbank anders ist als in der Datenquelle.

Bei Vergleichsanweisungen müssen Sie bei der Übergabe von Anweisungen an eine von der Groß-/Kleinschreibung unabhängige Datenquelle ebenfalls auf Fehlerquellen achten. Die den Zeichen 'I' und 'i' zugeordneten Wertigkeiten sind in einer Datenquelle, bei der die Groß-/Kleinschreibung nicht beachtet wird, gleich. Bei einer von der Groß-/Kleinschreibung unabhängigen Datenquelle mit einer englischen Codepage werden die Wörter **STEWART**, **StEWArT** und **stewart** als identisch interpretiert. Die zusammengeslossene DB2-Datenbank ist standardmäßig von der Groß-/Kleinschreibung abhängig und würde diesen Zeichen unterschiedliche Wertigkeiten zuordnen.

Sind die Sortierfolgen der zusammengeslossenen Datenbank und der Datenquelle unterschiedlich, überträgt der Server die Daten auf die zusammengeslossene Datenbank, sodass die Sortierung und der Vergleich lokal durchgeführt werden können. Durch das lokale Sortieren der Daten kommt der Server der Annahme der Benutzer entgegen, dass die Abfrageergebnisse nach der für den Server mit zusammengeslossenen Datenbanken festgelegten Sortierfolge sortiert werden.

Enthält die Abfrage ein Gleichheitszeichen, kann dieser Abschnitt der Abfrage im Pushdown-Modus weitergeleitet werden, auch wenn die Sortierfolgen nicht übereinstimmen (auf 'N' gesetzt). Das Vergleichselement C1 = 'A' könnte beispielsweise im Pushdown-Modus an eine Datenquelle weitergeleitet werden. Solche Abfragen können jedoch nicht im Pushdown-Modus weitergeleitet werden, wenn die Sortierfolge in der Datenquelle von der Groß-/Kleinschreibung unabhängig ist. Ist eine Datenquelle von der Groß-/Kleinschreibung unabhängig, werden für C1 = 'A' und C1 = 'a' dieselben Ergebnisse erzielt. Dies wird jedoch in einer von der Groß-/Kleinschreibung abhängigen Umgebung wie DB2 UDB nicht akzeptiert.

Administratoren können zusammengeslossene Datenbanken erstellen, deren Sortierfolge mit der Sortierfolge der Datenquelle übereinstimmt. Mit dieser Methode kann die Leistung beschleunigt werden, wenn alle Datenquellen dieselbe Sortierfolge verwenden oder die meisten bzw. alle Spaltenfunktionen für Datenquellen mit derselben Sortierfolge ausgeführt werden.

Das Abrufen von Daten für lokale Sortierungen und Vergleiche reduziert üblicherweise die Leistung. Daher sollte die zusammengeslossene Datenbank so konfiguriert werden, dass dieselben Sortierfolgen verwendet werden wie in den Datenquellen. Auf diese Weise kann die Leistung erhöht werden, da der Server mit zusammengeslossenen Datenbanken die Durchführung von Sortierungen und Vergleichen in Datenquellen zulässt. In DB2 für z/OS™ und OS/390® werden beispielsweise durch eine Klausel ORDER BY definierte Sortierungen durch eine auf einer EBCDIC-Codepage basierenden Sortierfolge implementiert. Soll der Server mit zusammengeslossenen Datenbanken Daten von DB2 für z/OS und OS/390 nach ORDER BY-Klauseln sortiert abrufen, sollte die zusammengeslossene Datenbank für die Verwendung einer auf einer EBCDIC-Codepage basierenden vordefinierten Sortierfolge konfiguriert werden.

Sind die Sortierfolgen in der zusammengeslossenen Datenbank und in der Datenquelle unterschiedlich und sollen die Daten in der Sortierfolge der Datenquelle sortiert werden, kann die Abfrage in einer Durchgriffssitzung übergeben oder die Abfrage in einer Sicht der Datenquelle angegeben werden.

Serveroptionen des Servers mit zusammengeschlossenen Datenbanken

Die oben aufgeführten, Pushdown-Möglichkeiten beeinflussenden Faktoren sind Merkmale des Datenbankservers, die nicht verändert werden können. Die folgenden Serveroptionen können verändert werden und beeinflussen in einigen Fällen möglicherweise die Abfrageleistung:

- **COLLATING_SEQUENCE.** Verfügt eine Datenquelle über eine andere Sortierfolge als die von DB2 für Linux, UNIX[®] und Windows[®], können von der Sortierfolge in DB2 abhängige Operationen nicht fern in einer Datenquelle ausgewertet werden. Ein Beispiel dafür ist das Ausführen von MAX-Spaltenfunktionen für eine Kurznamenzeichenspalte in einer Datenquelle mit einer anderen Sortierfolge. Da die Ergebnisse bei Auswertung der Funktion MAX in der fernen Datenquelle fehlerhaft sein können, führt die zusammengeschlossene Datenbank die Ergebnisoperation und die Funktion MAX lokal aus.
- **VARCHAR_NO_TRAILING_BLANKS.** Diese Option ist für Zeichenfolgen mit variabler Länge ohne folgende Leerzeichen bestimmt. Einige Datenquellen, wie z. B. Oracle, verwenden keine Vergleichssemantik mit Leerzeichen als Füllzeichen, wie beispielsweise DB2 für Linux, UNIX und Windows. Dieser Unterschied beim Auffüllen kann zu unerwarteten Ergebnissen führen. Wenn sichergestellt ist, dass alle Spalten VARCHAR und VARCHAR2 in einer Datenquelle keine folgenden Leerzeichen enthalten, kann das Erstellen dieser Serveroption für die Datenquelle in Betracht gezogen werden. Dabei müssen jedoch alle Objekte einschließlich Sichten berücksichtigt werden, die potenziell Kurznamen haben können.
- **DB2_MAXIMAL_PUSHDOWN.** Diese Option gibt die primären Bedingungen an, die das Abfrageoptimierungsprogramm bei der Auswahl eines Zugriffsplans verwendet. Das Abfrageoptimierungsprogramm kann Zugriffspläne entweder auf der Basis des Aufwands auswählen oder auf der Basis der Benutzerbestimmung, dass von den fernen Datenquellen so viel Abfrageverarbeitung wie möglich übernommen wird. Ist wenn DB2_MAXIMAL_PUSHDOWN auf 'Y' gesetzt, ist die Reduzierung des Datenaustauschs im Netz die vorrangige Bedingung für das Abfrageoptimierungsprogramm. Das Abfrageoptimierungsprogramm verwendet den Zugriffsplan, der die wenigsten Sendevorgänge an die Datenquelle ausführt. Wenn diese Serveroption auf 'Y' gesetzt ist, muss der Server mit zusammengeschlossenen Datenbanken einen Zugriffsplan verwenden, der möglicherweise mehr Aufwand als nötig verursacht. Wird ein anderer Zugriffsplan verwendet als der Plan, der den geringsten Aufwand verursacht, kann dies die Leistung verringern. Wenn eine gespeicherte Abfragetabelle (MQT = Materialized Query Table) auf dem Server mit zusammengeschlossenen Datenbanken eine Abfrage teilweise oder vollständig verarbeiten kann, kann ein Zugriffsplan verwendet werden, der die gespeicherte Abfragetabelle enthält. Wenn keine Operationen an die Datenquellen weitergeleitet werden, sondern stattdessen eine gespeicherte Abfragetabelle verwendet wird, reduziert dies den Datenaustausch im Netz. Ist die Serveroption DB2_MAXIMAL_PUSHDOWN auf 'Y' gesetzt, wird eine Abfrage, die in einem kartesischen Produkt resultiert, nicht im Pushdown-Modus an ferne Datenquellen weitergeleitet. Abfragen, die in einem kartesischen Produkt resultieren, werden von der zusammengeschlossenen Datenbank verarbeitet. Die Serveroption DB2_MAXIMAL_PUSHDOWN muss nicht auf 'Y' gesetzt werden, damit der Server mit zusammengeschlossenen Datenbanken die Verarbeitung von Abfragen im Pushdown-Modus an ferne Datenquellen sendet. Wenn diese Serveroption auf 'N' gesetzt ist (dies ist die Standardeinstellung), leitet das Abfrageoptimierungsprogramm Abfragen im Pushdown-Modus zur Verarbeitung an die Datenquellen weiter. Ist diese Option auf 'N' gesetzt, ist die wichtigste Bedingung für das Optimierungsprogramm jedoch der Aufwand und nicht der Datenaustausch im Netz.

Faktoren der Typen- und Funktionszuordnung

Die Standarddatentypzuordnungen sind in den Datenquellenwrappern integriert. Diese Zuordnungen sind so aufgebaut, dass alle Datentypen der Datenquelle über genügend Pufferspeicherbereich verfügen, um einen Laufzeitpufferüberlauf zu vermeiden. Die Typzuordnung kann für eine Datenquelle entsprechend den Anforderungen bestimmter Anwendungen angepasst werden. Greifen Sie beispielsweise in einer Oracle-Datenquelle auf eine Spalte mit dem Datentyp DATE zu, erfolgt standardmäßig die Zuordnung zum Datentyp TIMESTAMP von DB2 für Linux, UNIX und Windows. Sie können den lokalen Datentyp in den Datentyp DATE von DB2 für Linux, UNIX und Windows ändern. Durch diese Änderung wird die Verwendung einer Funktion SCALAR zur Extrahierung einer Untergruppe der gesamten im Datentyp TIMESTAMP gespeicherten Daten umgangen.

Die Standardfunktionszuordnungen sind ebenfalls in die Datenquellenwrapper integriert. Die zusammengesessene Datenbank kompensiert Funktionen, die von einer Datenquelle nicht unterstützt werden. In den folgenden drei Fällen werden Funktionen kompensiert:

- Eine Funktion ist in der Datenquelle nicht vorhanden. Einige Funktionen SYS-FUN sind beispielsweise in Datenquellen von DB2 für z/OS und OS/390 nicht vorhanden und erfordern daher lokale Kompensation.
- Eine Funktion ist in der Datenquelle zwar vorhanden, die Kenndaten des Operanden verletzen jedoch die Funktionseinschränkungen. Ein Beispiel hierfür ist der relationale Operator IS NULL. Die meisten Datenquellen unterstützen diesen Operator. In einigen Fällen gibt es jedoch Einschränkungen, sodass z. B. ein Spaltenname nur links neben dem Operator IS NULL zulässig ist.
- Eine fern ausgewertete Funktion kann ein anderes Ergebnis zurückgeben. Ein Beispiel hierfür ist der Operator '>' (größer als). Bei Datenquellen mit anderen Sortierfolgen gibt der Operator '>' (größer als) unter Umständen andere Ergebnisse zurück als bei der lokalen Auswertung durch DB2 für Linux, UNIX und Windows.

Zugehörige Konzepte:

- „Collating Sequences“ in *Application Development Guide: Programming Client Applications*
- „Datentypzuordnungen in einem System zusammengesessener Datenbanken“ auf Seite 53
- „Optimieren der Abfrageverarbeitung“ auf Seite 153
- „Pushdown-Analyse“ auf Seite 156
- „Kurznamenkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 163
- „Abfragekenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 165

Zugehörige Referenzen:

- Kapitel 21, „Serveroptionen für Systeme zusammengesessener Datenbanken“, auf Seite 253

Kurznamenkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten

Es gibt verschiedene kurznamenspezifische Faktoren, die Pushdown-Möglichkeiten beeinflussen können. Der lokale Datentyp einer Kurznamenspalte kann die vom Optimierungsprogramm ausgewertete Anzahl der Möglichkeiten in einer verknüpfenden Sortierung beeinflussen. Kurznamen können durch eine Spaltenoption markiert werden, um anzuzeigen, dass die Spalten keine folgenden Leerzeichen enthalten. Dadurch hat der SQL-Compiler die Möglichkeit, ein effizienteres Format für das Vergleichselement der an die Datenquellen gesendeten SQL-Anweisung zu generieren.

Lokaler Datentyp von Kurznamenspalten

Stellen Sie sicher, dass der lokale Datentyp einer Spalte nicht die Auswertung eines Vergleichselements in der Datenquelle verhindert. Die standardmäßigen Datentypzuordnungen sollen einen möglichen Überlauf vermeiden. Möglicherweise wird jedoch ein verknüpfendes Vergleichselement zwischen zwei Spalten unterschiedlicher Länge von der Datenquelle (deren verknüpfende Spalte kürzer ist) nicht berücksichtigt. Dies hängt davon ab, wie DB2[®] UDB die längere Spalte einbindet. Diese Situation kann die vom Optimierungsprogramm ausgewertete Anzahl der Möglichkeiten in einer verknüpfenden Sortierung beeinflussen. Spalten einer Oracle-Datenquelle beispielsweise, die unter Verwendung des Datentyps INTEGER oder INT erstellt wurden, erhalten den Datentyp NUMBER(38). Eine Kurznamenspalte für diesen Oracle-Datentyp erhält den lokalen Datentyp FLOAT, da in DB2 eine ganze Zahl im Bereich von 2^{*31} bis $(-2^{*31})-1$ liegt und dies in etwa dem Datentyp NUMBER(9) entspricht. In diesem Fall können Verknüpfungen zwischen einer Spalte ganzer Zahlen in DB2 und einer Spalte ganzer Zahlen in Oracle wegen der kürzeren verknüpfenden Spalte nicht in der DB2-Datenquelle erfolgen. Kann die Domäne dieser Oracle-Spalte ganzer Zahlen jedoch durch den DB2-Datentyp INTEGER erfasst werden, ändern Sie mit der Anweisung ALTER NICKNAME den lokalen Datentyp, sodass die Verknüpfung in der DB2-Datenquelle erfolgen kann.

Spaltenoptionen der zusammengeschlossenen Datenbank

Die Spaltenoptionen bewirken, dass der Wrapper die Daten in einer Spalte anders als sonst verarbeitet. Der SQL-Compiler und das Abfrageoptimierungsprogramm verwenden die Metadaten, um für die Daten bessere Zugriffspläne zu entwickeln. DB2 UDB behandelt das Objekt, auf das ein Kurzname verweist, wie eine Tabelle. Deshalb können für alle Datenquellenobjekte, für die ein Kurzname erstellt wird, Spaltenoptionen gesetzt werden. Die Anweisung ALTER NICKNAME kann zum Hinzufügen oder Ändern von Spaltenoptionen für Kurznamen verwendet werden. Es gibt zwei Spaltenoptionen:

- **NUMERIC_STRING.** Diese Spaltenoption gilt für Zeichentypspalten (CHAR und VARCHAR). Angenommen, eine Datenquelle hat eine andere Sortierfolge als die zusammengeschlossene Datenbank. Normalerweise sortiert der Server mit zusammengeschlossenen Datenbanken dann in der Datenquelle keine Spalten, die Zeichendaten enthalten. Stattdessen gibt er die Daten an die zusammengeschlossene Datenbank zurück und führt die Sortierung lokal aus. Angenommen, die Spalte ist ein Zeichendatentyp (CHAR und VARCHAR) und enthält ausschließlich numerische Zeichen ('0','1',..., '9'). Sie können dies durch die Zuordnung des Wertes 'Y' zu der Spaltenoption NUMERIC_STRING angeben. Somit kann das Abfrageoptimierungsprogramm von DB2 die Sortierung in der Datenquelle ausführen. Wird die Sortierung fern durchgeführt, kann der Systemaufwand für die Übertragung der Daten zum Server mit zusammengeschlossenen Datenbanken und die lokale Sortierung vermieden werden.

- `VARCHAR_NO_TRAILING_BLANKS`. Im Gegensatz zur gleichnamigen Serveroption kann diese Spaltenoption zur Angabe bestimmter Oracle-Spalten verwendet werden, die keine folgenden Leerzeichen enthalten. Der SQL-Compiler berücksichtigt diese Informationen dann bei der Pushdown-Analyse, wenn alle Operationen für Spalten mit dieser Einstellung überprüft werden. Der SQL-Compiler kann auf der Basis der Einstellung `VARCHAR_NO_TRAILING_BLANKS` ein anderes, aber äquivalentes Format für das Vergleichselement generieren, das in der an die Datenquelle gesendeten fernen SQL-Anweisung verwendet wird. Für die Datenquelle wird möglicherweise ein anderes Vergleichselement ausgewertet, das Endergebnis ist jedoch dasselbe.

Gespeicherte Abfragetabellen

Eine gespeicherte Abfragetabelle ist eine Übersichtstabelle, die auf der Basis der Ergebnismenge einer Abfrage erstellt wird. Die zum Erstellen einer gespeicherten Abfragetabelle verwendete Abfrage verwendet eine Gesamtauswahl, die eine `GROUP BY`-Klausel enthält, um Daten aus den Tabellen zusammenzufassen, auf die in der Abfrage verwiesen wird.

Im Gegensatz zu einer Sicht speichert eine gespeicherte Abfragetabelle die tatsächlichen Daten der Ergebnismenge in einer Tabelle. In einem System zusammengesetzter Datenbanken können Sie eine gespeicherte Abfragetabelle erstellen, die auf einen oder mehrere Kurznamen oder auf eine Kombination mehrerer Kurznamen und lokaler Tabellen verweist.

Eine gespeicherte Abfragetabelle, die von einer Abfrage definiert wird, die auf einen Kurznamen verweist, ist eine lokale Tabelle auf dem Server mit zusammengesetzten Datenbanken. Die gespeicherte Abfragetabelle enthält eine Kopie der Daten, die in den fernen Datenquellen gespeichert sind und die in der Ergebnismenge der Abfrage zurückgegeben wurden. Benutzer des Systems zusammengesetzter Datenbanken müssen sich nicht über das Vorhandensein der gespeicherten Abfragetabelle bewusst sein. Wenn eine Abfrage an den Server mit zusammengesetzten Datenbanken gesendet wird, die auf die Kurznamen verweist, auf denen die gespeicherte Abfragetabelle basiert, kann das Abfrageoptimierungsprogramm auf transparente Weise die lokale gespeicherte Abfragetabelle verwenden, anstatt auf die ferne Datenquelle zuzugreifen. Das Zugreifen auf lokale anstatt auf ferne Daten kann sich positiv auf die Leistung auswirken.

Gespeicherte Abfragetabellen können nur mit der Option `REFRESH DEFERRED` erstellt werden.

Zugehörige Konzepte:

- „Cachetabellen“ auf Seite 205
- „Optimieren der Abfrageverarbeitung“ auf Seite 153
- „Pushdown-Analyse“ auf Seite 156
- „Serverkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 158
- „Abfragekenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 165

Zugehörige Tasks:

- „Ändern eines Kurznamens“ auf Seite 39

Zugehörige Referenzen:

- Kapitel 24, „Kurznamenspaltenoptionen für Systeme zusammengesetzter Datenbanken“, auf Seite 283

Abfragekenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten

Eine Abfrage kann auf einen SQL-Operator verweisen, der Kurznamen von mehreren Datenquellen einschließt. Kombiniert der Server mit zusammengeschlossenen Datenbanken die Ergebnisse von zwei Datenquellen unter Verwendung eines einzigen Operators, muss die Operation auf dem Server mit zusammengeschlossenen Datenbanken ausgeführt werden. Ein Beispiel hierfür ist ein Gruppenoperator, z. B. UNION. Der Operator kann nicht direkt in einer fernen Datenquelle ausgewertet werden.

Zugehörige Konzepte:

- „Serverkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 158
- „Kurznamenkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 163

Entscheidungen bei der Pushdown-Analyse

Durch das Umschreiben von SQL-Anweisungen können bei der Verarbeitung von Abfragen durch den Server mit zusammengeschlossenen Datenbanken zusätzliche Pushdown-Möglichkeiten erzielt werden. Zur Unterstützung bei der Ermittlung von optimalen SQL-Umschreibungen werden in den folgenden Abschnitten verschiedene Tools vorgestellt, mit denen ermittelt werden kann, wo eine Abfrage für die Auslagerung (Pushdown) ausgewertet wird. Des Weiteren werden im Zusammenhang mit der Abfrageanalyse häufig gestellte Fragen (und im Hinblick darauf zu untersuchende Bereiche) aufgelistet und Themen zu Upgrades von Datenquellen behandelt.

Analysieren des Auswertungsortes einer Abfrage

Detaillierte Informationen des Abfrageoptimierungsprogramms befinden sich separat von dem eigentlichen Zugriffsplan in EXPLAIN-Tabellen. Mit Hilfe dieser Informationen ist eine intensive Analyse eines Zugriffsplans möglich. Auf die EXPLAIN-Tabellen kann von allen unterstützten Betriebssystemen zugegriffen werden. Sie enthalten Informationen über statische und dynamische SQL-Anweisungen. Sie können unter Verwendung von SQL-Anweisungen auf die EXPLAIN-Tabellen zugreifen. Damit werden eine einfache Bearbeitung der Ausgabe, der Vergleich verschiedener Abfragen und der Vergleich derselben Abfrage über einen bestimmten Zeitraum hinweg gewährleistet.

Vorgehensweise:

Es gibt zwei Möglichkeiten, Informationen zum globalen Zugriffsplan aus den EXPLAIN-Tabellen abzurufen.

- Formattool der EXPLAIN-Tabelle. Verwenden Sie das Tool **db2exfmt**, um die Informationen der EXPLAIN-Tabellen in einem vordefinierten Format darzustellen.

- Sie können die Tools **db2expln** und **dynexpln** verwenden, um den Zugriffsplan zu verstehen, der für eine bestimmte SQL-Anweisung ausgewählt wurde. Sie können auch die in die DB2®-Steuerzentrale integrierte EXPLAIN-Einrichtung zusammen mit Visual Explain verwenden, um den für eine bestimmte SQL-Anweisung ausgewählten Zugriffsplan zu verstehen. Mit EXPLAIN können sowohl dynamische als auch statische SQL-Anweisungen bearbeitet werden. Der einzige Unterschied zwischen EXPLAIN und Visual Explain ist, dass Visual Explain die EXPLAIN-Informationen in einem Grafikformat darstellt. Die Detaillierungsebene der beiden Methoden ist im Übrigen identisch. Zur vollständigen Verwendung der Ausgabe von **db2expln** und **dynexpln** müssen Sie mit Folgendem vertraut sein:
 - Den unterschiedlichen, unterstützten SQL-Anweisungen und der zu diesen Anweisungen gehörenden Terminologie (z. B. Vergleichselemente in einer Anweisung SELECT)
 - Dem Zweck eines Pakets (Zugriffsplan)
 - Dem Zweck und Inhalt der Systemkatalogtabellen
 - Allgemeinen Konzepten zur Anwendungsoptimierung

Ziehen Sie in Betracht, die EXPLAIN-Tools mit der Serveroption DB2_MAXIMAL_PUSHDOWN zu verwenden. Führen Sie die EXPLAIN-Tools mit der auf 'N' gesetzten Serveroption DB2_MAXIMAL_PUSHDOWN für eine Abfrage aus. Dies ist die Standardeinstellung für diese Option. Die Pushdown-Analyse bestimmt, welche Abschnitte der SQL im Pushdown-Modus weitergeleitet werden können. Das Abfrageoptimierungsprogramm generiert anschließend alle alternativen Pläne, die gegen keine der von der Pushdown-Analyse festgelegten Bedingungen verstoßen. Das Abfrageoptimierungsprogramm schätzt den Aufwand für jeden Plan und wählt den Plan mit dem geringsten Aufwand aus. Setzen Sie die Serveroption DB2_MAXIMAL_PUSHDOWN anschließend auf 'Y'. Verwenden Sie die EXPLAIN-Tools für dieselbe SQL-Anweisung. Der in der Ausgabe von EXPLAIN angezeigte Plan zeigt alle SQL-Operationen, die an die Datenquelle weitergeleitet werden können. Wenn Sie einen Unterschied zwischen den beiden Plänen feststellen, resultiert dieser Unterschied aus den Informationen, die das Abfrageoptimierungsprogramm zur Aufwandsoptimierung verwendet hat. Dazu gehören Indexierungsinformationen des Kurznamens, die Statistikdaten des Kurznamens und die Serverattribute.

Zugehörige Konzepte:

- „EXPLAIN-Tools“ in *Systemverwaltung: Optimierung*
- „SQL-EXPLAIN-Tools“ in *Systemverwaltung: Optimierung*
- „dynexpln“ in *Systemverwaltung: Optimierung*
- „Beschreibung der Ausgabe von db2expln und dynexpln“ in *Systemverwaltung: Optimierung*
- „Optimieren der Abfrageverarbeitung“ auf Seite 153
- „Pushdown-Analyse“ auf Seite 156
- „Entscheidungen für die Auswertung von Zugriffsplänen“ auf Seite 167
- „Aktualisieren und Anpassen von Datenquellen“ auf Seite 169

Zugehörige Tasks:

- „Globale Optimierung“ auf Seite 170

Zugehörige Referenzen:

- „db2exfmt - Explain Table Format Command“ in *Command Reference*
- „db2expln - SQL Explain Command“ in *Command Reference*

Entscheidungen für die Auswertung von Zugriffsplänen

In diesem Abschnitt werden typische Fragen zur Analyse des Zugriffsplans sowie zu untersuchende Bereiche zur Erweiterung von Pushdown-Möglichkeiten aufgeführt.

Warum wird dieses Vergleichselement nicht fern ausgewertet?

Diese Frage stellt sich bei einem selektiven Vergleichselement, das zum Filtern von Zeilen und somit zur Reduzierung des Netzwerkverkehrs verwendet werden könnte. Eine ferne Auswertung des Vergleichselements bestimmt auch, ob eine Verknüpfung von zwei Tabellen derselben Datenquelle fern ausgewertet werden kann.

Zu den zu untersuchenden Bereichen gehören:

- Vergleichselemente für Unterabfragen. Enthält dieses Vergleichselement eine Unterabfrage, die zu einer anderen Datenquelle gehört? Enthält dieses Vergleichselement eine Unterabfrage mit einem SQL-Operator, der von dieser Datenquelle nicht unterstützt wird? Nicht alle Datenquellen unterstützen Gruppenoperatoren in einem Vergleichselement.
- Funktionen von Vergleichselementen. Enthält dieses Vergleichselement eine Funktion, die von dieser fernen Datenquelle nicht ausgewertet werden kann? Relationale Operatoren werden als Funktionen klassifiziert.
- Einbindungsanforderungen von Vergleichselementen. Erfordert dieses Vergleichselement bei ferner Auswertung die Einbindung von Werten? Würde dies zur Verletzung von SQL-Einschränkungen in dieser Datenquelle führen?
- Globale Optimierung. Das Optimierungsprogramm hat festgestellt, dass eine lokale Verarbeitung weniger kostenintensiv ist.

Warum wird der Operator GROUP BY nicht fern ausgewertet?

Sie können folgende Bereiche überprüfen:

- Wird die Eingabe für den Operator GROUP BY fern ausgewertet? Wenn nicht, überprüfen Sie die Eingabe.
- Bestehen in der Datenquelle Einschränkungen für diesen Operator? Beispiele:
 - Begrenzte Anzahl von GROUP BY-Elementen
 - Begrenzte Anzahl der Bytes bei kombinierten GROUP BY-Elementen
 - Spaltenspezifikation nur für GROUP BY-Liste
- Unterstützt die Datenquelle diesen SQL-Operator?
- Globale Optimierung. Das Optimierungsprogramm hat festgestellt, dass eine lokale Verarbeitung weniger kostenintensiv ist.

Warum wird der Operator SET nicht fern ausgewertet?

Sie können folgende Bereiche überprüfen:

- Werden beide Operanden des Operators SET vollständig in derselben fernen Datenquelle ausgewertet? Wenn nicht, überprüfen Sie alle Operanden.
- Bestehen in der Datenquelle Einschränkungen für diesen Operator SET? Sind z. B. große Objekte oder lange Felder für diesen bestimmten Operator SET eine gültige Eingabe?

Warum wird die Operation ORDER BY nicht fern ausgewertet?

Überprüfen Sie Folgendes:

- Wird die Eingabe für den Operator ORDER BY fern ausgewertet? Wenn nicht, überprüfen Sie die Eingabe.
- Enthält die Klausel ORDER BY einen Zeichenausdruck? Wenn ja, verfügt die ferne Datenquelle über eine andere Sortierfolge als der Server mit zusammengeschlossenen Datenbanken?
- Bestehen in der Datenquelle Einschränkungen für diesen Operator? Ist beispielsweise die Anzahl der ORDER BY-Elemente begrenzt? Bestehen in der Datenquelle Einschränkungen in Bezug auf Spaltenspezifikationen für die Liste ORDER BY?

Warum wird eine ferne Anweisung INSERT mit Gesamtauswahl nicht vollständig fern ausgewertet?

Überprüfen Sie Folgendes:

- Könnte die Unterauswahl vollständig in der fernen Datenquelle ausgewertet werden? Wenn nicht, überprüfen Sie die Unterauswahl.
- Enthält die Unterauswahl einen Gruppenoperator? Wenn ja, unterstützt diese Datenquelle Gruppenoperatoren als Eingabe für eine Anweisung INSERT?
- Verweist die Unterauswahl auf die Zieltabelle? Wenn ja, lässt die Datenquelle diese Syntax zu?

Warum wird eine ferne Anweisung INSERT mit der Klausel VALUES nicht vollständig fern ausgewertet?

Überprüfen Sie Folgendes:

- Kann die Klausel VALUES vollständig in der fernen Datenquelle ausgewertet werden? Anders ausgedrückt, enthält ein Ausdruck eine Funktion, die von der fernen Datenquelle nicht unterstützt wird?
- Schließt der Ausdruck eine skalare Unterabfrage ein? Wird diese Syntax unterstützt?
- Verweist der Ausdruck auf die Zieltabelle? Wird diese Syntax unterstützt?

Warum wird eine ferne Anweisung UPDATE für eine Aktualisierung mit Suche nicht vollständig fern ausgewertet?

Überprüfen Sie Folgendes:

- Kann die Klausel SET vollständig in der fernen Datenquelle ausgewertet werden? Anders ausgedrückt, enthält ein Aktualisierungsausdruck eine Funktion, die von der fernen Datenquelle nicht unterstützt wird?
- Schließt die Klausel SET eine skalare Unterabfrage ein? Unterstützt die Datenquelle diese Syntax?
- Kann die Suchbedingung vollständig in der fernen Datenquelle ausgewertet werden? Wenn nicht, überprüfen Sie die Suchbedingung.
- Verweisen die Suchbedingung oder die Klausel SET auf die Zieltabelle? Unterstützt die Datenquelle diese Syntax?
- Verweisen die Suchbedingung oder die Klausel SET mit Korrelation auf die Zieltabelle? Unterstützt die Datenquelle diese Syntax?

Warum wird eine Anweisung UPDATE für eine positionierte Aktualisierung nicht vollständig fern ausgewertet?

Dies ist der Fall, wenn DB2[®] UDB den Aktualisierungsausdruck lokal auswertet und erst dann die Anweisung UPDATE an die Datenquelle sendet. Diese Methode beeinflusst die Leistung unerheblich.

- Kann die Klausel SET vollständig in der fernen Datenquelle ausgewertet werden? Anders ausgedrückt, enthält ein Aktualisierungsausdruck eine Funktion, die von der fernen Datenquelle nicht unterstützt wird?
- Schließt die Klausel SET eine skalare Unterabfrage ein? Unterstützt die Datenquelle diese Syntax?

Warum wird eine ferne Anweisung DELETE für einen gezielten Löschvorgang nicht vollständig fern ausgewertet?

Überprüfen Sie Folgendes:

- Kann die Suchbedingung vollständig in der fernen Datenquelle ausgewertet werden? Wenn nicht, überprüfen Sie die Suchbedingung.
- Verweist die Suchbedingung auf die Zieltabelle? Unterstützt die Datenquelle diese Syntax?
- Verweist die Suchbedingung mit Korrelation auf die Zieltabelle? Unterstützt die Datenquelle diese Syntax?

Zugehörige Konzepte:

- „Pushdown-Analyse“ auf Seite 156
- „Analysieren des Auswertungsortes einer Abfrage“ auf Seite 165

Aktualisieren und Anpassen von Datenquellen

Der DB2[®]-SQL-Compiler bezieht sich auf die im globalen Katalog gespeicherten Informationen über das SQL-Leistungsspektrum der jeweiligen Datenquelle. Diese Informationen müssen in regelmäßigen Abständen aktualisiert werden. Das SQL-Leistungsspektrum der Datenquellen kann sich mit neuen Versionen der einzelnen Datenquellen ändern. Werden Datenquellen aktualisiert oder angepasst, aktualisieren Sie die Informationen im globalen Katalog, um sicherzustellen, dass der SQL-Compiler die aktuellsten Informationen verwendet.

Verwenden Sie zum Aktualisieren des Katalogs die SQL-DDL-Anweisungen von DB2, wie z. B. CREATE FUNCTION MAPPING und ALTER SERVER.

Zugehörige Konzepte:

- „Optimieren der Abfrageverarbeitung“ auf Seite 153
- „Pushdown-Analyse“ auf Seite 156

Zugehörige Tasks:

- „Globale Optimierung“ auf Seite 170

Globale Optimierung

Der SQL-Compiler durchläuft zwei Phasen bei der Erstellung einer optimalen Zugriffsstrategie für die Auswertung einer auf eine ferne Datenquelle verweisende Abfrage. Bei diesen Phasen handelt es sich um die ferne SQL-Generierung und die globale Optimierung. Die Zugriffsstrategie für eine an die zusammengeschlossene Datenbank übergebene Abfrage kann daraus bestehen, die ursprüngliche Abfrage in mehrere Abfragefragmente zu zerlegen und dann die Ergebnisse zu kombinieren.

Das Abfrageoptimierungsprogramm verwendet die Ausgabe der Pushdown-Analysephase, um zu entscheiden, wo die Operation ausgewertet wird. Eine Operation kann lokal auf dem DB2-Server mit zusammengeschlossenen Datenbanken oder fern in der Datenquelle ausgewertet werden. Diese Entscheidung ist abhängig von der Ausgabe des ausgereiften Fixkostenmodells, das das Optimierungsprogramm verwendet. Dieses Modell ermittelt folgende Aufwände:

- Den Aufwand für die Auswertung der Operation.
- Den Aufwand für die Übertragung der Daten oder Nachrichten zwischen dem DB2-Server mit zusammengeschlossenen Datenbanken und den Datenquellen.

Ziel ist es, eine optimierte Abfrage zu generieren. Eine optimierte Abfrage ist eine Abfrage mit einem Zugriffsplan, der die Abfrageoperationen aller Datenquellen global im System zusammengeschlossener Datenbanken optimiert. *Globale Optimierung* ist dann erreicht, wenn ein Zugriffsplan mit dem geringsten Aufwand ausgewählt wird.

Der DB2-SQL-Compiler verfügt über eine Wissensbasis des Optimierungsprogramms, in der Daten über native Datenquellen enthalten sind. Das Optimierungsprogramm generiert keine fernen Zugriffspläne, die von bestimmten Datenbankverwaltungssystemen nicht generiert werden können. Anders ausgedrückt, das Optimierungsprogramm vermeidet das Generieren von Zugriffsplänen, die von Optimierungsprogrammen in fernen Datenquellen nicht interpretiert oder akzeptiert werden können.

Viele Faktoren können die Ausgabe der globalen Optimierung und somit auch die Abfrageleistung beeinflussen. Die Hauptfaktoren sind Serverkenndaten und Kurznamenkenndaten.

Zwischen relationalen und nicht relationalen Wrappern bestehen Unterschiede hinsichtlich der Details der Erstellung eines Zugriffsplans. Das Konzept und das Endergebnis sind jedoch identisch.

Zugehörige Konzepte:

- „Optimieren der Abfrageverarbeitung“ auf Seite 153
- „Pushdown-Analyse“ auf Seite 156
- „Serverkenndaten mit Auswirkungen auf die globale Optimierung“ auf Seite 171

Zugehörige Tasks:

- „Kurznamenkenndaten mit Auswirkungen auf die globale Optimierung“ auf Seite 173

Globale Optimierung - ausführliche Informationen

Sowohl Serverkenndaten als auch Kurznamenkenndaten haben Auswirkungen auf die globale Optimierung, Kurznamenkenndaten wirken sich jedoch stärker auf die vom Abfrageoptimierungsprogramm generierte Schätzung des anfallenden Aufwands aus.

Serverkenndaten mit Auswirkungen auf die globale Optimierung

Das Abfrageoptimierungsprogramm erhält durch die Einstellungen für die Serveroptionen Informationen über die Serverkenndaten der Datenquelle. Die Einstellungen für die Serveroptionen sind ein Teil der Serverdefinition der Datenquelle. Sie können bei der anfänglichen Erstellung der Serverdefinition die Einstellungen für die Serveroptionen in der Anweisung CREATE SERVER festlegen. Verwenden Sie die Anweisung ALTER SERVER, um zu einer vorhandenen Serverdefinition Serveroptionen hinzuzufügen. Die Einstellungen für die Serveroptionen werden im Katalog der zusammengeschlossenen Datenbank gespeichert.

Diese Optionen sind in Optionen für die Speicherposition (z. B. für den Rechnernamen der Datenquelle), Sicherheitsoptionen (z. B. Authentifizierungsinformationen) und Leistungsoptionen (z. B. das CPU-Verhältnis) unterteilt.

Mit Hilfe der Leistungsoptionen kann das Optimierungsprogramm ermitteln, ob Auswertungsoperationen in der Datenquelle ausgeführt werden können. Folgende Serveroptionen beeinflussen die Leistung, sodass eventuell eine Optimierung erforderlich ist:

- CPU_RATIO
- IO_RATIO
- COMM_RATE
- COLLATING_SEQUENCE
- PLAN_HINTS

| Gehen Sie bei der Optimierung der Serveroption CPU_RATIO, IO_RATIO oder
| COMM_RATE achtsam vor, da unerwartete Fehler auftreten können, wenn die
| Aufwandskalkulation für eine Abfrage Über- bzw. Unterläufe erzeugt.

Relatives Verhältnis der CPU-Geschwindigkeit

Dieser Wert gibt an, um wie viel höher oder niedriger die CPU-Geschwindigkeit der Datenquelle im Vergleich zur CPU-Geschwindigkeit von DB2[®] ist. Ein niedriges Verhältnis gibt an, dass die CPU der Workstation der Datenquelle schneller ist als die CPU der Workstation von DB2. Bei einem niedrigen Verhältnis versucht das Optimierungsprogramm möglicherweise, CPU-intensive Verarbeitungsoperationen im Pushdown-Modus an die Datenquelle weiterzuleiten. Ein niedriges Verhältnis ist ein Wert kleiner als 1.

| Die Einstellung 1 für die Serveroption CPU_RATIO 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. Wenn die
| Geschwindigkeit der CPU des DB2-Servers mit zusammengeschlossenen Datenban-
| ken um 50% unter der Geschwindigkeit der CPU der Datenquelle liegt, sollte die
| Serveroption CPU_RATIO den Wert .5 haben. Wenn die Geschwindigkeit der CPU

des DB2-Servers mit zusammengeschlossenen Datenbanken doppelt so hoch ist wie die Geschwindigkeit der CPU der Datenquelle, sollte die Serveroption CPU_RATIO den Wert 2 haben.

Relatives Verhältnis der E/A-Geschwindigkeit

Dieser Wert gibt an, um wie viel höher oder niedriger die E/A-Geschwindigkeit der Datenquelle im Vergleich zur E/A-Geschwindigkeit des Servers mit zusammengeschlossenen Datenbanken ist. Ein niedriges Verhältnis gibt an, dass die E/A-Geschwindigkeit der Workstation der Datenquelle höher ist als die E/A-Geschwindigkeit der Workstation von DB2. Bei einem niedrigen Verhältnis versucht das Optimierungsprogramm möglicherweise, E/A-intensive Verarbeitungsoperationen im Pushdown-Modus an die Datenquelle weiterzuleiten. Ein niedriges Verhältnis ist ein Wert kleiner als 1.

Die Einstellung 1 für die Serveroption IO_RATIO 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. Wenn die Geschwindigkeit des Ein-/Ausgabesystems des DB2-Servers mit zusammengeschlossenen Datenbanken um 50% unter der Geschwindigkeit des Ein-/Ausgabesystems der Datenquelle liegt, sollte die Serveroption IO_RATIO den Wert .5 haben. Wenn die Geschwindigkeit des Ein-/Ausgabesystems des DB2-Servers mit zusammengeschlossenen Datenbanken doppelt so hoch ist wie die Geschwindigkeit des Ein-/Ausgabesystems der Datenquelle, sollte die Serveroption IO_RATIO den Wert .5 haben.

Übertragungsgeschwindigkeit zwischen dem Server mit zusammengeschlossenen Datenbanken und der Datenquelle

Eine niedrige Übertragungsgeschwindigkeit gibt eine langsame Netzwerkkommunikation zwischen dem Server mit zusammengeschlossenen Datenbanken und der Datenquelle an. Bei niedrigen Übertragungsgeschwindigkeiten reduziert das Abfrageoptimierungsprogramm die Anzahl der Nachrichten, die an die bzw. von der betreffenden Datenquelle gesendet werden. Ist die Serveroption COMM_RATE sehr niedrig eingestellt, erstellt das Optimierungsprogramm eine Abfrage, die minimalen Netzwerkverkehr erfordert.

Sortierfolge der Datenquelle

Die Auswahl der Sortierfolge kann die Leistung der zusammengeschlossenen Datenbank beeinflussen. Verwenden Sie die Serveroption COLLATING_SEQUENCE, um anzugeben, ob die Sortierfolge einer Datenquelle mit der lokalen Sortierfolge der zusammengeschlossenen DB2-Datenbank übereinstimmt. DB2 UDB kann sortierungsabhängige Verarbeitungsoperationen mit Zeichendaten im Pushdown-Modus an die Datenquelle weiterleiten. Stimmen die Sortierfolgen der Datenquelle und der zusammengeschlossenen Datenbank nicht überein, betrachtet das Optimierungsprogramm die von dieser Datenquelle abgerufenen Daten als unsortiert. DB2 UDB ruft dann die relevanten Daten ab und führt alle sortierungsabhängigen Verarbeitungsoperationen von Zeichendaten lokal aus. Hierdurch kann die Leistung gemindert werden. Im Abschnitt zu den die *Pushdown-Möglichkeiten beeinflussenden Serverkenndaten* finden Sie ausführliche Informationen zu Sortierfolgen.

Ferne Planhinweise

Verwenden Sie die Serveroption `PLAN_HINTS`, um ferne Planhinweise zu generieren. 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.

Sie sollten einige Tests durchführen, um festzustellen, ob diese Serveroption die Leistung Ihrer Abfragen verbessert.

Sie können in einer Abfrage keine eigenen Planhinweise codieren.

Ist die Funktion für Planhinweise aktiviert, enthält die an die Datenquelle gesendete Abfrage zusätzliche Informationen. Eine Anweisung, die an ein Oracle-Optimierungsprogramm mit Planhinweisen gesendet wird, könnte z. B. wie folgt aussehen:

```
SELECT /*+ INDEX (table1, t1index)*/  
  col1  
FROM table1
```

Der Planhinweis ist die Zeichenfolge `/*+ INDEX (table1, t1index)*/`

Zugehörige Konzepte:

- „Optimieren der Abfrageverarbeitung“ auf Seite 153
- „Serverkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 158

Zugehörige Tasks:

- „Ändern von Serverdefinitionen und Serveroptionen“ auf Seite 32
- „Globale Optimierung“ auf Seite 170
- „Kurznamenkenndaten mit Auswirkungen auf die globale Optimierung“ auf Seite 173

Zugehörige Referenzen:

- Kapitel 21, „Serveroptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 253

Kurznamenkenndaten mit Auswirkungen auf die globale Optimierung

Es gibt verschiedene kurznamenspezifische Faktoren, die die globale Optimierung beeinflussen können. Dazu gehören die Indexinformationen und die Statistikdaten des globalen Katalogs.

Die Indexinformationen und die Statistikdaten des globalen Katalogs, die dem SQL-Compiler zur Verfügung stehen, müssen aktuell sein.

Indexspezifikationen

Der SQL-Compiler verwendet zur Optimierung von Abfragen Indexinformationen. Eine Datenquellentabelle erhält nur dann Indexinformationen, wenn der Kurzname für diese Tabelle erstellt wird. Nach der Erstellung des Kurznamens werden die Indexinformationen für diese Datenquellentabelle auf dem Server mit zusammengeschlossenen Datenbanken nicht aktualisiert. Wenn sich die fernen Indexinformationen ändern, können Sie die auf dem Server mit zusammengeschlossenen Datenbanken gespeicherten Indexinformationen ändern, indem Sie den Kurznamen für die Tabelle löschen und erneut erstellen. Wenn ein neuer Index für die Datenquellentabelle hinzugefügt wurde, können Sie alternativ auf dem Server mit zusammengeschlossenen Datenbanken eine Indexspezifikation für die Tabelle definieren.

Für Kurznamen von Objekten, die keine Indizes haben (wie z. B. Sichten, Synonyme oder Objekte von nicht relationalen Datenquellen), werden keine Indexinformationen erfasst.

Hat ein Objekt mit einem Kurznamen keinen Index, können Sie dafür eine Indexspezifikation erstellen. Indexspezifikationen bilden eine Indexdefinition im globalen Katalog. Die Indexspezifikation selbst ist kein Index. Verwenden Sie die Anweisung `CREATE INDEX` mit der Klausel `SPECIFICATION ONLY`, um eine Indexspezifikation zu erstellen. Die Syntax zum Erstellen einer Indexspezifikation für einen Kurznamen ist ähnlich wie die Syntax zum Erstellen einer Indexspezifikation für eine lokale Tabelle.

Ziehen Sie in folgenden Fällen das Erstellen von Indexspezifikationen in Betracht:

- Eine Tabelle erhält einen neuen Index.
- Sie erstellen einen Kurznamen für ein Datenquellenobjekt, das keine Indizes enthält, wie z. B. eine Sicht oder ein Synonym.

Bevor Sie Anweisungen des Typs `CREATE INDEX...SPECIFICATION ONLY` für einen Kurznamen für eine Datenquellensicht eingeben, sollten Sie sich über Ihre Anforderungen bewusst sein:

- Handelt es sich bei der fernen Sicht um eine einfache Anweisung `SELECT` für eine Datenquellentabelle mit Index, kann das Erstellen einer Indexspezifikation für den Kurznamen, die mit dem Index für die Datenquellentabelle übereinstimmt, die Abfrageleistung erheblich verbessern.
- Wenn eine Indexspezifikation für eine ferne Sicht erstellt wird, die nicht aus einfachen `SELECT`-Anweisungen besteht (z. B. eine durch die Verknüpfung zweier Tabellen erstellte Sicht), wird die Abfrageleistung unter Umständen beeinträchtigt.

Angenommen, es wird eine Indexspezifikation für eine ferne Sicht erstellt, die eine Verknüpfung zweier Tabellen ist. Das Optimierungsprogramm wählt unter Umständen diese Sicht als inneres Element einer Verknüpfung mit Verschachtelungsschleife aus. Die Abfrage verfügt möglicherweise über ein schlechtes Leistungsverhalten, da die Verknüpfung mehrmals ausgewertet wird. Eine alternative Methode besteht darin, für jede Tabelle, auf die in der Sicht der Datenquelle verwiesen wird, einen Kurznamen zu erstellen und dann eine Sicht für zusammengeschlossene Datenbanken zu erstellen, die auf beide Kurznamen verweist.

Statistikdaten des globalen Katalogs

Die zusammengeschlossene Datenbank verwendet Katalogstatistiken für Objekte mit Kurznamen, um die Abfrageverarbeitung zu optimieren. Diese Statistiken werden aus der Datenquelle abgerufen, wenn Sie unter Verwendung der Anweisung CREATE NICKNAME für ein Datenquellenobjekt einen Kurznamen erstellen. Die zusammengeschlossene Datenbank überprüft die Existenz des Objekts in der Datenquelle 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 zu Datenquellen u. U. vom Optimierungsprogramm verwendet werden können, empfiehlt es sich, Statistiken (unter Verwendung des Datenquellenbefehls, der RUNSTATS entspricht) in der Datenquelle zu aktualisieren, bevor ein Kurzname erstellt wird.

Die Katalogstatistik beschreibt die Gesamtgröße der Tabellen und Sichten sowie den Wertebereich in zugeordneten Spalten. Die abgerufenen Informationen umfassen Folgendes:

- Die Anzahl Zeilen in einem Kurznamenobjekt
- Die Anzahl Seiten, die ein Kurzname belegt
- Die Anzahl eindeutiger Werte in jeder Spalte einer Tabelle
- Die Anzahl eindeutiger Werte in Spalten eines Index
- Die höchsten/niedrigsten Werte einer Spalte

Die zusammengeschlossene Datenbank kann zwar die statistischen Daten einer Datenquelle abrufen, sie kann jedoch Aktualisierungen von vorhandenen statistischen Daten in Datenquellen nicht automatisch ermitteln. Außerdem verfügt die zusammengeschlossene Datenbank über keinen Mechanismus zur Bearbeitung von Objektdefinitionen oder zur Ausführung von strukturellen Änderungen der Objekte in der Datenquelle (wie z. B. beim Hinzufügen einer Spalte zu einer Tabelle).

Wenn sich die statistischen Daten oder die strukturellen Kenndaten eines fernen Objekts, für das ein Kurzname definiert ist, ändern, haben Sie drei Auswahlmöglichkeiten für die Aktualisierung der Statistik:

- Verwenden Sie die Aktualisierungsfunktion für Kurznamenstatistiken in der DB2-Steuerzentrale.
- Führen Sie das RUNSTATS entsprechende Programm in der Datenquelle aus. Löschen Sie anschließend den aktuellen Kurznamen, und erstellen Sie den Kurznamen erneut. Dies ist die empfohlene Methode zur Aktualisierung von Statistikdaten.
- Aktualisieren Sie die Statistik in der Katalogsicht SYSSTAT.TABLES manuell. Verwenden Sie diese Methode nur dann, wenn Sie wissen, dass die Statistikdaten in der fernen Datenquelle falsch oder unvollständig sind.

Zeilenänderungen aktualisieren:

Wurden in der Datenquelle eine große Anzahl Zeilen hinzugefügt oder gelöscht, bemerkt die zusammengeschlossene Datenbank diese Änderungen nicht. Die Leistung kann sich jedoch verringern, da das Optimierungsprogramm Entscheidungen auf Grund von nicht mehr genauen Kurznameninformationen trifft. Aktualisieren Sie die Statistikdaten des Kurznamens, sodass das Optimierungsprogramm bei der Entwicklung von Zugriffsplänen für die Verarbeitung von Abfragen für die Datenquelle über genaue Statistikdaten verfügt.

Spaltenänderungen aktualisieren:

Werden in der Datenquelle Spalten hinzugefügt, gelöscht oder geändert, erhalten Sie unter Umständen fehlerhafte Ergebnisse oder eine Fehlermeldung. Angenommen, Sie haben den Kurznamen *EUROSALES*, der auf die Tabelle *europa* in einer Sybase-Datenbank verweist. Wird der Tabelle eine neue Spalte mit dem Namen *CZECH* hinzugefügt, wird diese von der zusammengeschlossenen Datenbank nicht erkannt. Für Abfragen, die auf diese Spalte verweisen, wird eine Fehlermeldung ausgegeben.

Werden in einem Datenquellenobjekt Spalten geändert, müssen mehrere Schritte ausgeführt werden, um die Statistikdaten für dieses Objekt im Katalog der zusammengeschlossenen Datenbank zu aktualisieren:

1. Führen Sie in der Datenquelle das DB2 RUNSTATS entsprechende Dienstprogramm aus. Dadurch werden die im Katalog der Datenquelle gespeicherten Statistikdaten aktualisiert.
2. Löschen Sie den aktuellen Kurznamen für das Datenquellenobjekt mit der Anweisung DROP NICKNAME.
3. Erstellen Sie den Kurznamen mit der Anweisung CREATE NICKNAME neu.

Der Kurzname hat die Statistikdaten nun aktualisiert, sodass sie mit dem Objektschema der Datenquelle konsistent ist.

Zugehörige Konzepte:

- „Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht“ auf Seite 219
- „Kurznamenkenndaten mit Auswirkungen auf die Pushdown-Möglichkeiten“ auf Seite 163
- „Serverkenndaten mit Auswirkungen auf die globale Optimierung“ auf Seite 171

Zugehörige Tasks:

- „Löschen eines Kurznamens“ auf Seite 51
- „Erstellen von Indexspezifikationen für Datenquellenobjekte“ auf Seite 84

Entscheidungen der globalen Optimierung

In den folgenden Abschnitten werden verschiedene Tools vorgestellt, die zur Analyse der Abfrageoptimierung verwendet werden können. Des Weiteren werden im Zusammenhang mit der Abfrageoptimierung häufig gestellte Fragen (und im Hinblick darauf zu untersuchende Bereiche) behandelt.

Analysieren der globalen Optimierung

Detaillierte Informationen des Abfrageoptimierungsprogramms befinden sich separat von dem eigentlichen Zugriffsplan in EXPLAIN-Tabellen. Mit Hilfe dieser Informationen ist eine intensive Analyse eines Zugriffsplans möglich. Auf die EXPLAIN-Tabellen kann von allen unterstützten Betriebssystemen zugegriffen werden. Sie enthalten Informationen über statische und dynamische SQL-Anweisungen. Sie können unter Verwendung von SQL-Anweisungen auf die EXPLAIN-Tabellen zugreifen. Damit werden eine einfache Bearbeitung der Ausgabe, der Vergleich verschiedener Abfragen und Vergleiche derselben Abfrage über einen bestimmten Zeitraum hinweg gewährleistet.

Vorgehensweise:

Es gibt zwei Möglichkeiten, Informationen zum globalen Zugriffsplan aus den EXPLAIN-Tabellen abzurufen.

- Formattool der EXPLAIN-Tabelle. Verwenden Sie das Tool **db2exfmt**, um die Informationen der EXPLAIN-Tabellen in einem vordefinierten Format darzustellen.
- Sie können die Tools **db2expln** und **dynexpln** verwenden, um den Zugriffsplan zu verstehen, der für eine bestimmte SQL-Anweisung ausgewählt wurde. Sie können auch die in die DB2®-Steuerzentrale integrierte EXPLAIN-Einrichtung zusammen mit Visual Explain verwenden, um den für eine bestimmte SQL-Anweisung ausgewählten Zugriffsplan zu verstehen. Mit EXPLAIN können sowohl dynamische als auch statische SQL-Anweisungen bearbeitet werden. Der einzige Unterschied zwischen EXPLAIN und Visual Explain ist, dass Visual Explain die EXPLAIN-Informationen in einem Grafikformat darstellt. Die Detaillierungsebene der beiden Methoden ist im Übrigen identisch. Zur vollständigen Verwendung der Ausgabe von **db2expln** und **dynexpln** müssen Sie mit Folgendem vertraut sein:
 - Den unterschiedlichen, unterstützten SQL-Anweisungen und der zu diesen Anweisungen gehörenden Terminologie (z. B. Vergleichselemente in einer Anweisung SELECT)
 - Dem Zweck eines Pakets (Zugriffsplan)
 - Dem Zweck und Inhalt der Systemkatalogtabellen
 - Allgemeinen Konzepten zur Anwendungsoptimierung

Zugehörige Konzepte:

- „EXPLAIN-Tools“ in *Systemverwaltung: Optimierung*
- „SQL-EXPLAIN-Tools“ in *Systemverwaltung: Optimierung*
- „dynexpln“ in *Systemverwaltung: Optimierung*
- „Beschreibung der Ausgabe von db2expln und dynexpln“ in *Systemverwaltung: Optimierung*
- „Optimieren der Abfrageverarbeitung“ auf Seite 153
- „Durch den Zugriffsplan bestimmte Möglichkeiten der Optimierung“ auf Seite 178

Zugehörige Tasks:

- „Globale Optimierung“ auf Seite 170

Zugehörige Referenzen:

- „db2exfmt - Explain Table Format Command“ in *Command Reference*
- „db2expln - SQL Explain Command“ in *Command Reference*

Durch den Zugriffsplan bestimmte Möglichkeiten der Optimierung

In diesem Abschnitt werden typische Fragen zur Optimierung sowie zu untersuchende Bereiche zur Verbesserung der Leistung aufgeführt.

Warum wird eine Verknüpfung zwischen zwei Kurznamen derselben Datenquelle fern ausgewertet?

Zu den zu untersuchenden Bereichen gehören:

- Verknüpfungsoperationen. Kann die Datenquelle Verknüpfungsoperationen unterstützen?
- Vergleichselemente für Verknüpfungen. Kann das Vergleichselement für die Verknüpfung in der fernen Datenquelle ausgewertet werden? Wenn nicht, überprüfen Sie das Vergleichselement für die Verknüpfung.
- Anzahl der Zeilen im Ergebnis der Verknüpfung. Die Anzahl der Zeilen kann mit Visual Explain ermittelt werden. Ist die durch die Verknüpfung erzeugte Zeilenmenge sehr viel größer als die Kombination der beiden Kurznamen? Sind die Zahlen realistisch? Wenn nicht, ziehen Sie die Aktualisierung der Kurznamenstatistik unter Verwendung des Dienstprogramms RUNSTATS in Betracht.

Warum wird der Operator GROUP BY nicht fern ausgewertet?

Zu den zu untersuchenden Bereichen gehören:

- Syntax des Operators. Überprüfen Sie, ob der Operator in der fernen Datenquelle ausgewertet werden kann.
- Anzahl der Zeilen. Überprüfen Sie unter Verwendung von Visual Explain die in der Ein- und Ausgabe des Operators GROUP BY geschätzte Anzahl der Zeilen. Liegen diese beiden Zahlen nah beieinander? Wenn ja, wertet das Optimierungsprogramm diesen Operator GROUP BY aus Gründen der Effizienz eventuell lokal aus. Sind diese beiden Zahlen realistisch? Wenn nicht, ziehen Sie die Aktualisierung der Kurznamenstatistik unter Verwendung von RUNSTATS in Betracht.

Warum wird die Anweisung nicht vollständig fern ausgewertet?

Das Optimierungsprogramm führt eine aufwandsabhängige Optimierung durch. Auch wenn die Pushdown-Analyse angibt, dass alle Operatoren in der fernen Datenquelle ausgewertet werden können, nimmt das Optimierungsprogramm seine Aufwandsschätzung als Basis für die Generierung eines global optimalen Zugriffsplans. Dieser Zugriffsplan hängt von vielen Faktoren ab. Angenommen, die ferne Datenquelle kann alle Operationen in der ursprünglichen Abfrage verarbeiten. Die CPU-Geschwindigkeit der fernen Datenquelle ist jedoch sehr viel niedriger als die des Servers mit zusammengeschlossenen Datenbanken. In diesem Fall kann es von Vorteil sein, die Verarbeitungen stattdessen auf dem DB2[®]-Server mit zusammengeschlossenen Datenbanken auszuführen. Sind die Ergebnisse nicht zufriedenstellend, überprüfen Sie die Serverstatistik in der Katalogtabelle SYSSTAT.SERVEROPTIONS.

Warum verfügt ein vom Optimierungsprogramm generierter und vollständig fern ausgewerteter Zugriffsplan über eine schlechtere Leistung als die ursprüngliche, direkt in der fernen Datenquelle ausgeführte Abfrage?

Zu den zu untersuchenden Bereichen gehören:

- Die vom DB2-Abfrageoptimierungsprogramm generierte ferne SQL-Anweisung. Stellen Sie sicher, dass diese mit der ursprünglichen Abfrage identisch ist. Überprüfen Sie eventuelle Änderungen in der Sortierung des Vergleichselements. Ein gutes Abfrageoptimierungsprogramm sollte unabhängig von der Sortierfolge des Vergleichselements einer Abfrage arbeiten. Nicht alle DBMS-Optimierungsprogramme sind jedoch identisch. Das Optimierungsprogramm der fernen Datenquelle generiert auf der Basis der Sortierfolge der Vergleichselemente für die Eingabe einen anderen Zugriffsplan. Trifft dies zu, ist dieses Problem dem fernen Optimierungsprogramm zuzuordnen. Sie können die Sortierfolge der Vergleichselemente für die Eingabe in DB2 UDB ändern oder das Serviceunternehmen der fernen Datenquelle um Unterstützung bitten.

Überprüfen Sie auch die Ersetzung von Vergleichselementen. Ein gutes Abfrageoptimierungsprogramm sollte unabhängig von äquivalenten Ersetzungen von Vergleichselementen arbeiten. Möglicherweise generiert das Optimierungsprogramm der fernen Datenquelle auf der Basis der Vergleichselemente für die Eingabe einen anderen Zugriffsplan. Einige Optimierungsprogramme können beispielsweise keine transitiven Schließungsanweisungen für Vergleichselemente generieren.

- Anzahl der zurückgegebenen Zeilen. Sie können diese Zahl von Visual Explain abrufen. Gibt die Abfrage eine große Zeilenanzahl zurück, kann im Netzwerkverkehr ein Engpass auftreten.
- Zusätzliche Funktionen. Enthält die ferne SQL-Anweisung im Vergleich zur ursprünglichen Abfrage zusätzliche Funktionen? Einige dieser Zusatzfunktionen können zum Konvertieren von Datentypen generiert werden. Stellen Sie sicher, dass diese Funktionen erforderlich sind.

Zugehörige Konzepte:

- „Pushdown-Analyse“ auf Seite 156
- „Entscheidungen für die Auswertung von Zugriffsplänen“ auf Seite 167
- „Analysieren der globalen Optimierung“ auf Seite 176

Zugehörige Tasks:

- „Globale Optimierung“ auf Seite 170

Systemmonitorelemente, die die Leistung beeinflussen

Der DB2-Datenbanksystemmonitor stellt statistische Informationen zum aktuellen Status des Datenbankmanagers, Aktivitätsinformationen, wie z. B. Zähler, und andere Maße der Datenbankverarbeitung zusammen.

In einem System zusammengeschlossener Datenbanken können Sie den DB2-Datenbanksystemmonitor dazu verwenden, um Informationen zur Datenbankaktivität, zur Systemleistung und zur Anwendungsleistung zusammenzustellen.

Der Zeitmarkenmonitorschalter wird verwendet, um die Antwortzeiten der Interaktionen zwischen der zusammengeschlossenen Datenbank und einer Datenquelle zu verfolgen.

Die folgenden zusammengesetzten Datenelemente werden durch den Zeitmarkenschalter verfolgt:

- Antwortzeit zum Erstellen eines Kurznamens
- Antwortzeit zum Löschen
- Antwortzeit zum Einfügen
- Zeit für einen Durchgriff
- Antwortzeit für Abfragen
- Zeit für ferne Sperren
- Antwortzeit zum Aktualisieren

Die Standardeinstellung für den Zeitmarkenmonitorschalter ist ON.

Empfehlung: Sie können die Leistung erhöhen, indem Sie die Einstellung für den Zeitmarkenmonitorschalter für alle Anwendungen in OFF ändern. Wenn der Zeitmarkenschalter für eine Anwendung auf ON gesetzt ist, fährt das System mit dem Erfassen der Antwortzeiten fort. Sie verbessern die Leistung deshalb nicht, wenn Sie den Zeitmarkenschalter nur für einige Ihrer Anwendungen ausschalten.

Das Ausschalten des Schalters hat andere Auswirkungen.

- Wenn Sie den Zeitmarkenmonitorschalter für alle Anwendungen ausschalten, müssen Sie das DB2-Exemplar stoppen und erneut starten, um die Änderung zu implementieren.
- Das Ausschalten des Zeitmarkenmonitorschalters inaktiviert das Zusammenstellen von Zeitmarkeninformationen für Anwendungen mit und ohne zusammengesetzte Datenbanken. Auch die lokale Datenbank empfängt keine Zeitmarkeninformationen.

Wenn Sie Zeitmarkeninformationen für lokale Anwendungen ohne zusammengesetzte Datenbanken benötigen, sollten Sie den Zeitmarkenmonitorschalter nicht ausschalten.

Sie können den Zeitmarkenschalter für alle Anwendungen auf OFF setzen, indem Sie den folgenden Befehl verwenden:

```
update dbm cfg using dft_mon_timestamp off
```

Setzen Sie danach folgende Befehle ab:

```
db2stop  
db2start
```

Durch das Stoppen und Starten von DB2 UDB wird sichergestellt, dass der Schalter für alle Anwendungen ausgeschaltet ist.

Spezifische Informationen zu den Elementen, die durch den Zeitmarkenschalter protokolliert werden, werden in einem separaten Abschnitt besprochen.

Zugehörige Referenzen:

- Kapitel 31, „Systemmonitorelemente der zusammengesetzten Datenbanken“, auf Seite 339

Kapitel 12. Parallelität bei Abfragen, die auf Kurznamen verweisen

In diesem Kapitel wird beschrieben, wie Parallelität bei Abfragen funktioniert, die auf Kurznamen verweisen.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Parallelität bei Abfragen, die auf Kurznamen verweisen“
- „Partitionsinterne Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 182
- „Aktivieren der partitionsinternen Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 183
- „Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 183
- „Aktivieren von partitionsübergreifender Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 186
- „Rechenpartitionsgruppen“ auf Seite 186
- „Definieren einer Rechenpartitionsgruppe“ auf Seite 187
- „Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen - erwartetes Leistungsverhalten“ auf Seite 188
- „Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 188
- „Aktivieren von gemischter Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 189
- „Parallele Zugriffspläne von Abfragen, die auf Kurznamen verweisen“ auf Seite 189
- „Partitionsinterne Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne“ auf Seite 190
- „Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne“ auf Seite 192
- „Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne“ auf Seite 196

Parallelität bei Abfragen, die auf Kurznamen verweisen

Abfragen, die Kurznamen enthalten, können an drei Arten der abfrageinternen Parallelität beteiligt sein:

- Partitionsinterne Abfrageparallelität bei Einzelpartitionskonfigurationen mit Multiprozessoren
- Partitionsübergreifende Abfrageparallelität bei Mehrpartitionskonfigurationen
- Gemischte Abfrageparallelität, die aus partitionsinterner und -übergreifender Parallelität besteht, wobei jede Partition auf einem SMP-Computer ausgeführt wird

Zugehörige Konzepte:

- „Parallelität“ in *Systemverwaltung: Konzept*
- „Partitions- und Prozessorumgebungen“ in *Systemverwaltung: Konzept*

- „Das DB2-Prozessmodell“ in *Systemverwaltung: Optimierung*
- „Rechenpartitionsgruppen“ auf Seite 186
- „Parallele Zugriffspläne von Abfragen, die auf Kurznamen verweisen“ auf Seite 189
- „Partitionsinterne Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 182
- „Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 183
- „Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 188

Zugehörige Referenzen:

- Kapitel 20, „Wrapperoptionen für Systeme zusammenschlossener Datenbanken“, auf Seite 251

Partitionsinterne Parallelität bei Abfragen, die auf Kurznamen verweisen

Die partitionsinterne Parallelität bezieht sich auf den Prozess der Teilung einer Abfrage in mehrere gleichzeitig ausführbare Teile, die von mehreren Prozessen in einer einzelnen Datenbankpartition parallel ausgeführt werden. Bei Abfragen für zusammenschlossene Datenbanken kann der Teil einer Abfrage mit lokalen Daten parallel ausgeführt werden, während der Teil mit Kurznamen seriell ausgeführt wird.

Abfragen, die auf lokale Tabellen und Kurznamen verweisen, können schneller als in den vorherigen Releases von DB2[®] Information Integrator ausgeführt werden, da mehrere Prozessoren die lokalen Teile der Abfrage bearbeiten können.

Der Datenbankkonfigurationsparameter DFT_DEGREE und das Sonderregister CURRENT DEGREE steuern den Grad der partitionsinternen Parallelität.

Zugehörige Konzepte:

- „Parallelität“ in *Systemverwaltung: Konzept*
- „Partitions- und Prozessorumgebungen“ in *Systemverwaltung: Konzept*

Zugehörige Tasks:

- „Aktivieren der partitionsinternen Parallelität für Abfragen“ in *Systemverwaltung: Implementierung*
- „Aktivieren der partitionsinternen Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 183

Zugehörige Referenzen:

- „Partitionsinterne Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne“ auf Seite 190

Aktivieren der partitionsinternen Parallelität bei Abfragen, die auf Kurznamen verweisen

Für Abfragen, die auf lokale Tabellen und Kurznamen in einer Multiprozessorumgebung verweisen, können Sie die partitionsinterne Parallelität aktivieren. Der Server mit zusammengeschlossenen Datenbanken kann dann die lokalen Tabellen parallel verarbeiten.

Einschränkungen:

Das System zusammengeschlossener Datenbanken kann nur den lokalen Teil einer Abfrage parallel verarbeiten. Die Koordinatorpartition verarbeitet alle Operationen für den fernen Teil einer Abfrage seriell.

Vorgehensweise:

1. Setzen Sie den Datenbankkonfigurationsparameter `INTRA_PARALLEL` auf YES (JA).
2. Setzen Sie den Datenbankkonfigurationsparameter `MAX_QUERYDEGREE` auf einen Wert, der größer als 1 ist.
3. Setzen Sie den Datenbankkonfigurationsparameter `DFT_DEGREE` auf einen Wert, der größer als 1 ist, oder setzen Sie das Sonderregister `CURRENT_DEGREE`.

Wenn Sie den Parameter `DFT_DEGREE` auf ANY (Beliebig) setzen, ist die Standardstufe der partitionsinternen Parallelität gleich der Anzahl der Prozessoren auf dem Computer.

Zugehörige Konzepte:

- „Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 181

Zugehörige Referenzen:

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

Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen

Die partitionsübergreifende Parallelität bezieht sich auf den Prozess der Teilung einer einzelnen Abfrage in mehrere Teile, die in verschiedenen Partitionen einer partitionierten Datenbank parallel ausgeführt werden.

Bei Abfragen, die auf lokale und ferne Daten verweisen, kann der Server mit zusammengeschlossenen Datenbanken die fernen Daten an jede lokale Partition verteilen. In Abb. 7 auf Seite 184 sehen Sie das Konzept der partitionsübergreifenden Parallelität, die sowohl lokale als auch ferne Datenquellen einbezieht. Der obere Teil der Abbildung zeigt, wie dieser Abfragetyp in früheren Releases von DB2® Information Integrator verarbeitet wurde. Die fernen Kurznamendaten und die lokalen partitionierten Daten wurden seriell in einer einzigen Koordinatorpartition verarbeitet. Der untere Teil der Abbildung zeigt, wie DB2 Information Integrator diese Abfragetypen im aktuellen Release verarbeitet. Die Datenbank verteilt die Kurznamendaten zur parallelen Verarbeitung an die Partitionen des lokalen Systems.

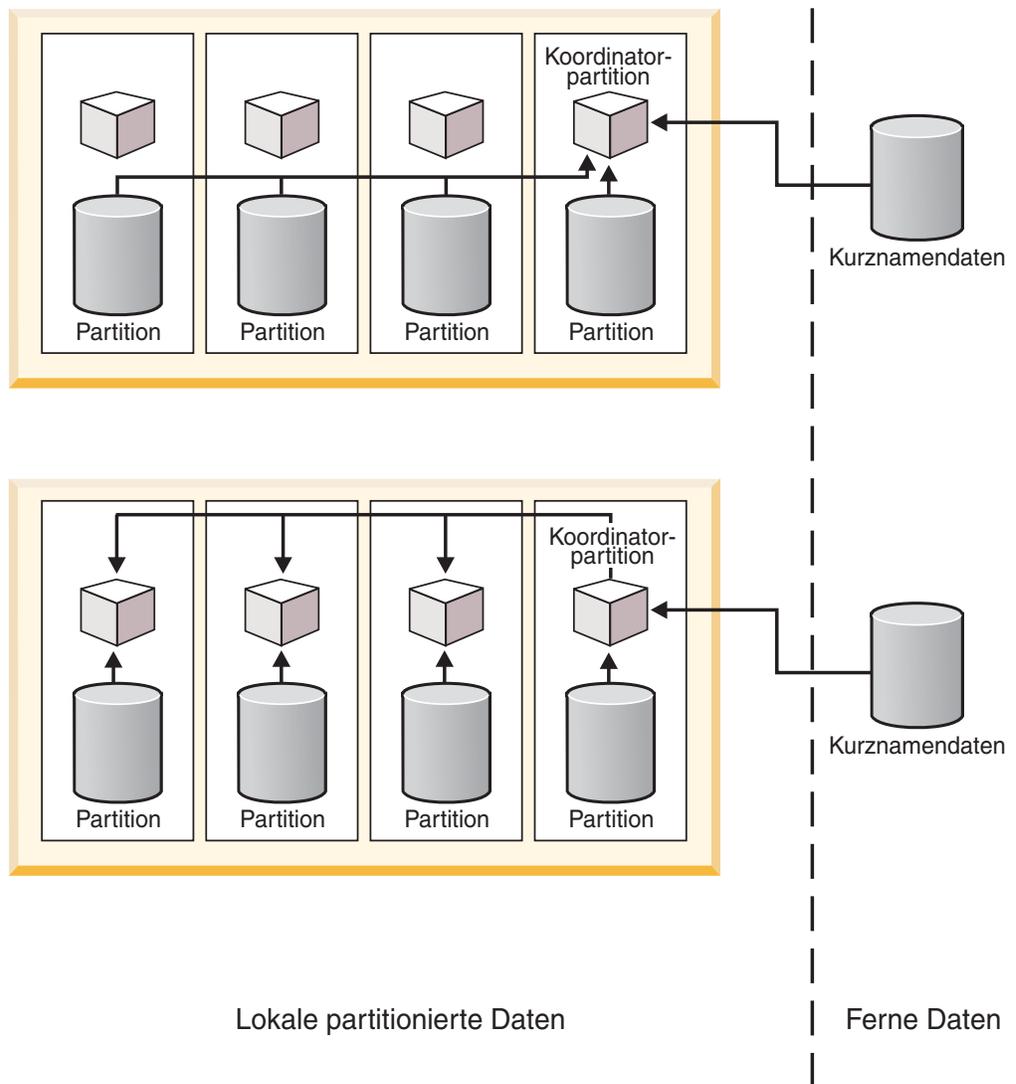


Abbildung 7. Partitionsübergreifende Parallelität lokaler und ferner Datenquellen

In Abb. 8 auf Seite 185 sehen Sie das Konzept der partitionsübergreifenden Parallelität, an der nur ferne Datenquellen beteiligt sind. Der obere Teil der Abbildung zeigt die serielle Verarbeitung der fernen Kurznamendaten in einer einzigen Koordinatorpartition. Der untere Teil der Abbildung zeigt, wie die Koordinatorpartition die Daten über eine Rechenpartitionsgruppe verteilt.

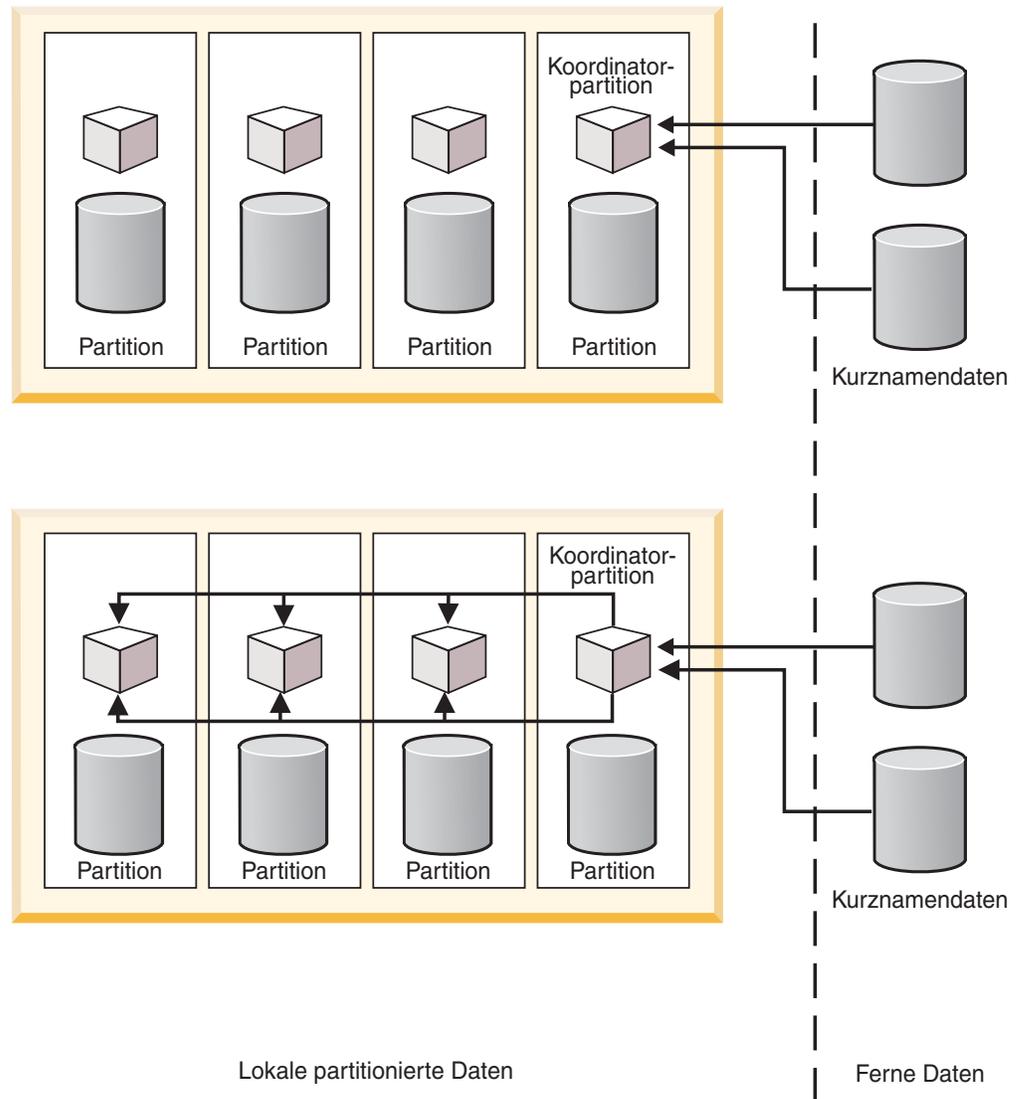


Abbildung 8. Partitionsübergreifende Parallelität einer Abfrage, die nur auf ferne Datenquellen verweist

Zugehörige Konzepte:

- „Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 181
- „Rechenpartitionsgruppen“ auf Seite 186

Zugehörige Tasks:

- „Aktivieren von partitionsübergreifender Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 186

Aktivieren von partitionsübergreifender Parallelität bei Abfragen, die auf Kurznamen verweisen

Abfragen, die auf Kurznamen verweisen, die über mehrere Partitionen parallel ausgeführt werden können, können schneller ausgeführt werden.

In einer partitionierten Datenbankumgebung können Abfragen für zusammengesetzte Datenbanken unter folgenden Bedingungen parallel ausgeführt werden:

- Wenn sie eine Kombination aus Kurznamen und lokalen partitionierten Tabellen beinhalten.
- Wenn sie nur Kurznamen beinhalten, die eine Rechenpartitionsgruppe verwenden.

Sie müssen für die partitionsübergreifende Parallelität in einer partitionierten Umgebung keine Datenbankparameter bzw. Datenbankkonfigurationsparameter definieren.

Einschränkungen:

Es können nur die Teile einer Abfrage, die auf Kurznamen mit abgeschirmten Wrappern verweisen, parallel ausgeführt werden. Die Teile einer Abfrage, die auf Kurznamen mit gesicherten Wrappern verweisen, können nicht parallel ausgeführt werden.

Vorgehensweise:

1. Konfigurieren Sie eine Rechenpartitionsgruppe.
2. Setzen Sie die Anweisung CREATE WRAPPER bzw. ALTER WRAPPER ab, und setzen Sie die Option DB2_FENCED auf Y.

Zugehörige Konzepte:

- „Rechenpartitionsgruppen“ auf Seite 186
- „Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 183

Rechenpartitionsgruppen

Eine Rechenpartitionsgruppe definiert eine Reihe von Partitionen, die das Optimierungsprogramm zur Durchführung einer dynamischen Umverteilungsoperation für Verknüpfungsoperationen verwendet. Eine Rechenpartitionsgruppe ist eine Datenbankpartitionsgruppe (nicht IBM-CATNODEGROUP), die im Systemkatalog SYSCAT.DBPARTITIONGROUPS angegeben wird.

Sie können Rechenpartitionsgruppen für die partitionsübergreifende Abfrageparallelität für zusammengesetzte Datenbanken verwenden. Wenn Sie eine Rechenpartitionsgruppe verwenden, können Sie die Teile von Abfragen, in denen Kurznamen verwendet werden, parallel ausführen. Bei einem Abfrageplan, der eine Rechenpartitionsgruppe umfasst, werden vom Server mit zusammengesetzten Datenbanken Kurznamendaten über die Partitionen umverteilt, um eine parallele Verknüpfung zu erstellen. Mit Hilfe eines solchen Plans können Abfragen schneller ausgeführt werden, wenn in der Verknüpfung ein hohes Volumen an Kurznamendaten verarbeitet wird.

Verwenden Sie die Registrierdatenbankvariable DB2_COMPPARTITIONGROUP, um die Rechenpartitionsgruppe anzugeben.

Zugehörige Konzepte:

- „Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 181

Zugehörige Tasks:

- „Definieren einer Rechenpartitionsgruppe“ auf Seite 187

Definieren einer Rechenpartitionsgruppe

Die Definition einer Rechenpartitionsgruppe ermöglicht es dem Optimierungsprogramm, einen Plan zu verwenden, mit dem Kurznamendaten an die Partitionen der Rechenpartitionsgruppe verteilt werden. Sie definieren eine Rechenpartitionsgruppe, um die partitionsübergreifende Abfrageparallelität für Abfragen oder Teile von Abfragen zu aktivieren, die nur auf Kurznamen verweisen.

Voraussetzungen:

Alle Partitionsgruppen, die zur Darstellung der Rechenpartitionsgruppe in allen Datenbanken des Exemplars verwendet werden, müssen denselben Namen haben. Sie können diese Partitionsgruppen zwar in jeder Datenbank unterschiedlich definieren, der Name muss jedoch identisch sein.

Einschränkungen:

Das Optimierungsprogramm verwendet Rechenpartitionsgruppen nur für die Teile einer Abfrage, die auf Kurznamen verweisen, ohne dabei auf lokale Daten zu verweisen.

Vorgehensweise:

Geben Sie in einer DB2-Befehlszeile den folgenden Befehl ein, um eine Rechenpartitionsgruppe zu definieren:

```
db2set DB2_COMPPARTITIONGROUP=partitiongroup_name
```

Dabei ist `partitiongroup_name` der Name der Partitionsgruppe, die Sie als Rechenpartitionsgruppe definieren möchten.

Im folgenden Beispiel sehen Sie, wie die Rechenpartitionsgruppe FINANCE3 mit der Registrierdatenbankvariablen DB2_COMPPARTITIONGROUP definiert wird.

```
db2set DB2_COMPPARTITIONGROUP=FINANCE3
```

Zugehörige Konzepte:

- „Rechenpartitionsgruppen“ auf Seite 186

Zugehörige Referenzen:

- „db2set - DB2 Profile Registry Command“ in *Command Reference*

Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen - erwartetes Leistungsverhalten

Bei Abfragen, die auf eine Kombination lokaler partitionierter Tabellen und Kurznamen verweisen, kann das Optimierungsprogramm einen Ausführungsplan auswählen, der Kurznamendaten auf die geeigneten Partitionen umverteilt. Mit Hilfe von Umverteilungsplänen können Abfragen schneller ausgeführt werden, wenn die Menge an Kurznamendaten in der Verknüpfung kleiner als die Menge an lokalen partitionierten Daten ist. Wenn die Menge der Kurznamendaten in der Verknüpfung erheblich größer als die Menge der lokalen Daten ist, ist die Verwendung eines parallelen Plans zur Umverteilung der Kurznamendaten unwahrscheinlich. Wenn das Optimierungsprogramm keinen parallelen Plan auswählt, führt der Server mit zusammengeschlossenen Datenbanken die Verknüpfungen zwischen Kurznamen und lokalen Tabellen in der Koordinatorpartition seriell durch.

Bei Verknüpfungen zwischen zwei Kurznamen kann ein Ausführungsplan von Vorteil sein, bei dem die Daten unter allen Partitionen einer Rechenpartitionsgruppe verteilt werden, sofern hierbei eine große Datenmenge verarbeitet wird. Die Vorteile einer parallelen Verarbeitung von Verknüpfungen mit großen Datenmengen überwiegt den zusätzlichen Aufwand der Umverteilung der Daten über mehrere Partitionen. Wenn die Menge an Kurznamendaten relativ gering ist, ist die Verknüpfung nicht aufwendig genug, dass sich der Extraaufwand einer Umverteilung der Daten über die Partitionen lohnt. Im Allgemeinen wählt das Optimierungsprogramm Rechenpartitionsgruppenpläne aus, wenn die Menge der beteiligten Kurznamendaten groß ist. Andernfalls verknüpft der Server mit zusammengeschlossenen Datenbanken die Kurznamen in der Koordinatorpartition seriell.

Zugehörige Konzepte:

- „Rechenpartitionsgruppen“ auf Seite 186
- „Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 183

Zugehörige Referenzen:

- „Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne“ auf Seite 192

Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen

Bei Abfragen mit lokalen Tabellen und Kurznamen in einer partitionierten Umgebung können Sie sowohl mit partitionsinterner als auch mit partitionsübergreifender Parallelität arbeiten. Der Server mit zusammengeschlossenen Datenbanken kann ferne Daten auf Partitionen verteilen und Daten parallel in jeder Partition verarbeiten.

Zugehörige Konzepte:

- „Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 181

Zugehörige Tasks:

- „Aktivieren von gemischter Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 189

Aktivieren von gemischter Parallelität bei Abfragen, die auf Kurznamen verweisen

Abfragen für zusammengeschlossene Datenbanken, die parallel ausgeführt werden, können schneller ausgeführt werden. Mit Hilfe der partitionsinternen und der partitionsübergreifenden Parallelität können Sie die Leistung von Abfragen verbessern, die auf lokale und ferne Daten verweisen. Abfragen für zusammengeschlossene Datenbanken verweisen auf eine Kombination aus Kurznamen und lokalen partitionierten Tabellen.

Vorgehensweise:

1. Setzen Sie den Datenbankkonfigurationsparameter `MAX_QUERYDEGREE` auf einen Wert, der größer als 1 ist.
2. Setzen Sie den Datenbankkonfigurationsparameter `DFT_DEGREE` auf einen Wert, der größer als 1 ist, oder setzen Sie das Sonderregister `CURRENT DEGREE`. Wenn Sie den Parameter `DFT_DEGREE` auf `ANY` (Beliebig) setzen, ist die Standardstufe der partitionsinternen Parallelität gleich der Anzahl der SMP-Prozessoren auf dem Computer.
3. Konfigurieren Sie eine Rechenpartitionsgruppe.
4. Geben Sie die Anweisung `CREATE WRAPPER` bzw. `ALTER WRAPPER` ein, und setzen Sie die Option `DB2_FENCED` auf `Y`.

Zugehörige Konzepte:

- „Rechenpartitionsgruppen“ auf Seite 186
- „Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 188

Zugehörige Tasks:

- „Trusted and fenced mode process environments“ in *IBM DB2 Information Integrator Wrapper Developer's Guide*

Zugehörige Referenzen:

- „Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne“ auf Seite 196

Parallele Zugriffspläne von Abfragen, die auf Kurznamen verweisen

Die `SQL-EXPLAIN`-Einrichtung erfasst Informationen zu den Zugriffsplänen, die das Optimierungsprogramm zur Verarbeitung von Abfragen verwendet.

Zugehörige Konzepte:

- „Die `SQL-EXPLAIN`-Einrichtung“ in *Systemverwaltung: Optimierung*
- „`EXPLAIN`-Tools“ in *Systemverwaltung: Optimierung*
- „`SQL-EXPLAIN`-Tools“ in *Systemverwaltung: Optimierung*

Zugehörige Referenzen:

- „Partitionsinterne Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne“ auf Seite 190
- „Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne“ auf Seite 192
- „Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne“ auf Seite 196

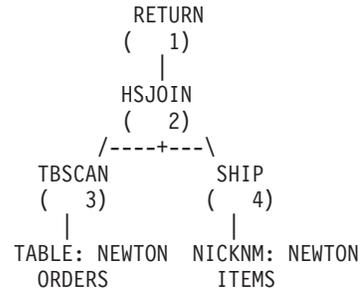
Partitionsinterne Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne

Die EXPLAIN-Einrichtung von DB2 UDB kann den Zugriffsplan generieren, den das Optimierungsprogramm bei der Abfrageverarbeitung verwendet. Die folgenden Beispiele zeigen, wie das Optimierungsprogramm in einer Umgebung mit partitionsinterner Parallelität auf Kurznamendaten zugreift.

Beispiel 1: Ohne Parallelitätsunterstützung von DB2 Information Integrator Version 8.2

Bei diesem Beispiel verarbeitet der Server mit zusammengeschlossenen Datenbanken die Verknüpfung zwischen der lokalen Tabelle ORDERS und dem Kurznamen ITEMS seriell. Die partitionsinterne Parallelität wird nicht verwendet.

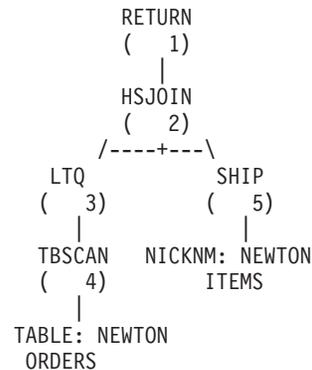
```
SELECT *
FROM ORDERS A, ITEMS B
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```



Beispiel 2: Mit Parallelitätsunterstützung von DB2 Information Integrator Version 8.2

Bei diesem Beispiel einer Verknüpfung kann die Abfrage schneller ausgeführt werden, da die lokale Tabelle vor der seriellen Verknüpfung mit dem Kurznamen parallel eingelesen wird.

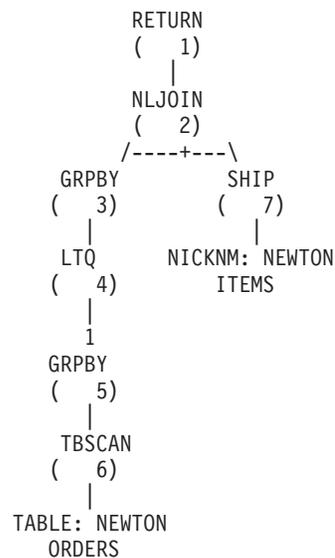
```
SELECT *
FROM ORDERS A, ITEMS B
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```



Beispiel 3: Partitionsinterne Parallelität bei Spaltenberechnung

Bei diesem Beispiel werden die lokalen Tabellendaten von der Datenbank in der Partition parallel zusammengefasst, wodurch die Ausführung der Spaltenberechnung verbessert wird. Die Verknüpfung der lokalen Tabelle und des Kurznamens erfolgt seriell im Koordinatorknoten.

```
SELECT *
FROM ITEMS A
WHERE ID =
  (SELECT MAX(ID)
   FROM ORDERS
   WHERE NUMBER = 10)
```



Zugehörige Konzepte:

- „Beispiel 2: Zugriffsplan für Einzelpartition mit partitionsinterner Parallelität“ in *Systemverwaltung: Optimierung*
- „Parallele Zugriffspläne von Abfragen, die auf Kurznamen verweisen“ auf Seite 189

Zugehörige Tasks:

- „Aktivieren der partitionsinternen Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 183

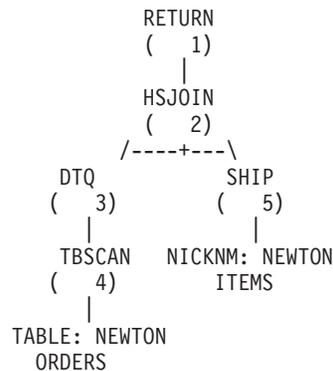
Partitionsübergreifende Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne

Die EXPLAIN-Einrichtung von DB2 UDB kann den Zugriffsplan generieren, den das Optimierungsprogramm bei der Abfrageverarbeitung verwendet. Die folgenden Beispiele zeigen, wie das Optimierungsprogramm in einer Umgebung mit partitionsübergreifender Parallelität auf Kurznamendaten zugreift.

Beispiel 1: Gesicherter Modus

In diesem Beispiel verwendet der Kurzname einen gesicherten Wrapper. Die Datenbank führt die Verknüpfung zwischen der lokalen Tabelle und dem Kurznamen in der Koordinatorpartition seriell durch. Die Datenbank sendet die über zwei Partitionen verteilten lokalen Daten an die Koordinatorpartition. Der Server mit zusammengeschlossenen Datenbanken verknüpft dann die lokalen Daten mit den Kurznamendaten. Die Datenbank verknüpft Kurznamen, die mit Hilfe eines gesicherten Wrappers in der Koordinatorpartition definiert wurden, seriell. Die Datenbank kann die Daten nicht über mehrere Partitionen verteilen, um eine parallele Verknüpfung zu erstellen.

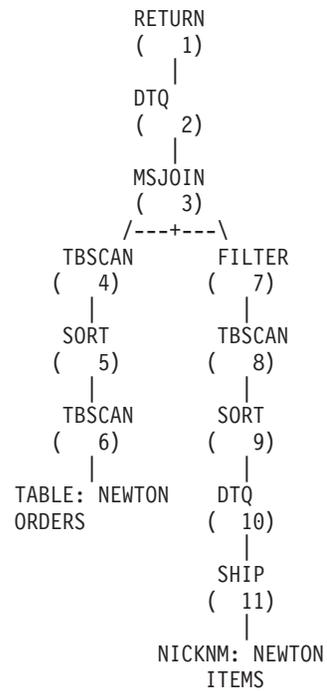
```
SELECT *  
FROM ORDERS A, ITEMS B  
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```



Beispiel 2: Abgeschirmter Modus

In diesem Beispiel verwendet der Kurzname einen abgeschirmten Wrapper. Der Server mit zusammengeschlossenen Datenbanken verteilt die Kurznamendaten an andere Partitionen und führt die Verknüpfung mit den lokalen Daten parallel durch.

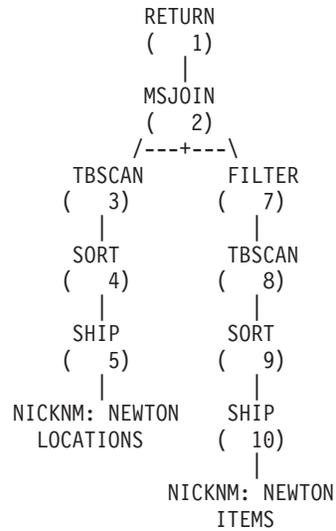
```
SELECT *  
FROM ORDERS A, ITEMS B  
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```



Beispiel 3: Abgeschirmter Modus ohne Rechenpartitionsgruppe

In diesem Beispiel verwenden die beiden Kurznamen einen abgeschirmten Wrapper; eine Rechenpartitionsgruppe wird nicht definiert. Der Server mit zusammengeschlossenen Datenbanken führt die Verknüpfung in der Koordinatorpartition durch. Der Server mit zusammengeschlossenen Datenbanken verteilt die Daten für die Verarbeitung nicht an andere Partitionen.

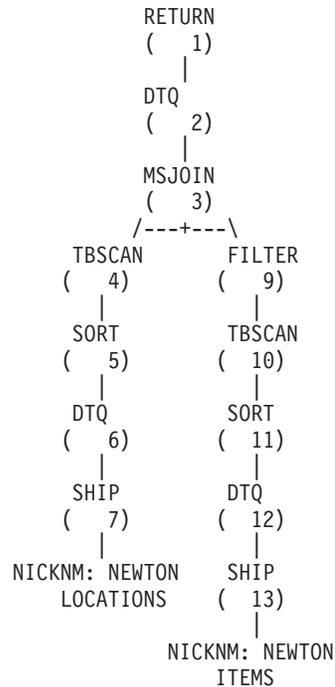
```
SELECT *  
FROM ITEMS A, LOCATIONS B  
WHERE A.ID1 = B.ID1
```



Beispiel 4: Abgeschirmter Modus mit Rechenpartitionsgruppe

In diesem Beispiel verwenden die Kurznamen abgeschirmte Wrapper; eine Rechenpartitionsgruppe wird definiert. In diesem Fall wählt das Optimierungsprogramm einen Plan aus, bei dem die Daten von der Koordinatorpartition an die anderen Partitionen der Rechenpartitionsgruppe verteilt werden.

```
SELECT *  
FROM ITEMS A, LOCATIONS B  
WHERE A.ID = B.ID
```



Zugehörige Konzepte:

- „Beispiel 3: Zugriffsplan für mehrere Partitionen mit partitionsübergreifender Parallelität“ in *Systemverwaltung: Optimierung*
- „Parallele Zugriffspläne von Abfragen, die auf Kurznamen verweisen“ auf Seite 189

Zugehörige Tasks:

- „Trusted and fenced mode process environments“ in *IBM DB2 Information Integrator Wrapper Developer's Guide*
- „Aktivieren von partitionsübergreifender Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 186

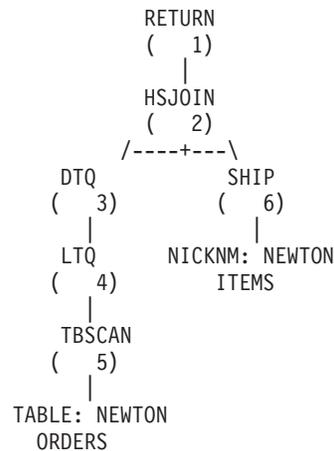
Gemischte Parallelität bei Abfragen, die auf Kurznamen verweisen - Beispiele für Zugriffspläne

Die EXPLAIN-Einrichtung von DB2 UDB kann den Zugriffsplan erstellen, den das Optimierungsprogramm bei der Abfrageverarbeitung verwendet. Die folgenden Beispiele zeigen, wie das Optimierungsprogramm in einer Umgebung mit partitionsinterner und partitionsübergreifender Parallelität auf Kurznamendaten zugreift.

Beispiel 1: Gesicherter Modus

Das folgende Beispiel zeigt eine Verknüpfung zwischen einer lokalen Tabelle und einem Kurznamen im gesicherten Modus. Der Server mit zusammengeschlossenen Datenbanken verarbeitet die lokalen Daten in jeder Partition parallel, bevor er diese Daten mit den Kurznamendaten in der Koordinatorpartition verknüpft. Der Server mit zusammengeschlossenen Datenbanken verarbeitet die Kurznamendaten nicht parallel auf den verschiedenen Partitionen oder den Prozessoren der einzelnen Partitionen.

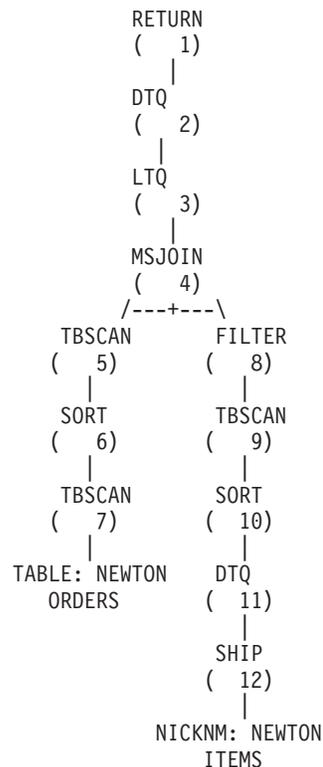
```
SELECT *  
FROM ORDERS A, ITEMS B  
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```



Beispiel 2: Abgeschirmter Modus

Das folgende Beispiel zeigt eine Verknüpfung zwischen einer lokalen Tabelle und einem Kurznamen im abgeschirmten Modus. Der Server mit zusammengeschlossenen Datenbanken verteilt die Kurznamendaten von der Koordinatorpartition aus an die anderen Partitionen im System. Der Server mit zusammengeschlossenen Datenbanken verarbeitet die lokalen Tabellendaten und die Kurznamendaten in den Partitionen und Prozessoren parallel.

```
SELECT *
FROM ORDERS A, ITEMS B
WHERE A.ID1 = B.ID1 AND B.ITEM = 3
```



Zugehörige Konzepte:

- „Beispiel 4: Zugriffsplan für mehrere Partitionen mit partitionsübergreifender und partitionsinterner Parallelität“ in *Systemverwaltung: Optimierung*
- „Parallele Zugriffspläne von Abfragen, die auf Kurznamen verweisen“ auf Seite 189

Zugehörige Tasks:

- „Trusted and fenced mode process environments“ in *IBM DB2 Information Integrator Wrapper Developer's Guide*
- „Aktivieren von gemischter Parallelität bei Abfragen, die auf Kurznamen verweisen“ auf Seite 189

Kapitel 13. Gespeicherte Abfragetabellen und Systeme zusammengeschnittener Datenbanken

In diesem Kapitel wird beschrieben, wie Sie gespeicherte Abfragetabellen verwenden können, die auf Kurznamen in Ihrem System zusammengeschnittener Datenbanken verweisen.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Gespeicherte Abfragetabellen und Systeme zusammengeschnittener Datenbanken - Übersicht“
- „Erstellen einer gespeicherten Abfragetabelle in Systemen zusammengeschnittener Datenbanken“ auf Seite 200
- „Datenquellenspezifische Einschränkungen für gespeicherte Abfragetabellen“ auf Seite 201
- „Einschränkungen bei der Verwendung gespeicherter Abfragetabellen mit Kurznamen“ auf Seite 204

Gespeicherte Abfragetabellen und Systeme zusammengeschnittener Datenbanken - Übersicht

Eine gespeicherte Abfragetabelle ist eine Tabelle, die die Ergebnisse einer Abfrage im Cache speichert. Wenn Sie die Abfrage erneut übergeben, kann die Datenbanksteuerkomponente die Daten aus der gespeicherten Abfragetabelle zurückgeben. Sie können gespeicherte Abfragetabellen mit Kurznamen verwenden, um die Leistung einer Abfrage zu verbessern und um eine Logikkomponente einzubinden. Bei der Erstellung von Cachetabellen verwenden Sie ebenfalls gespeicherte Abfragetabellen.

Das SQL-Optimierungsprogramm stellt fest, ob eine Abfrage mit einer gespeicherten Abfragetabelle effizienter ausgeführt werden kann als mit den Basistabellen oder Kurznamen. Das Optimierungsprogramm berücksichtigt für die Auswahl der gespeicherten Abfragetabelle die folgenden Faktoren:

- Die gespeicherte Abfragetabelle muss teilweise oder ganz mit der Abfrage übereinstimmen.
- Die Bedingung für das Aktualisierungsalter muss erfüllt werden.
- Der Zugriffsplan, der eine gespeicherte Abfragetabelle verwendet, muss kostengünstiger als der Zugriffsplan sein, der die Basistabellen oder Kurznamen verwendet.

Die folgenden Datenquellen unterstützen gespeicherte Abfragetabellen:

- Relationale Datenquellen
 - DRDA[®]
 - Informix[®]
 - ODBC
 - Oracle
 - Sybase
 - MS SQL Server
 - Teradata

- Nicht relationale Datenquellen
 - BioRS
 - BLAST
 - Documentum
 - Entrez
 - Excel
 - HMMER
 - IBM® Lotus® Extended Search
 - Dateien mit Tabellenstruktur
 - Web-Services
 - WebSphere® Business Integration
 - XML

Zugehörige Konzepte:

- „Cachetabellen“ auf Seite 205
- „Performance and tuning planning— materialized query tables in a federated system“ in *IBM DB2 Information Integrator Application Developer’s Guide*

Zugehörige Tasks:

- „Erstellen einer gespeicherten Abfragetabelle“ in *Systemverwaltung: Implementierung*
- „Ändern der Merkmale einer gespeicherten Abfragetabelle“ in *Systemverwaltung: Implementierung*
- „Erstellen einer gespeicherten Abfragetabelle in Systemen zusammengesetzter Datenbanken“ auf Seite 200

Zugehörige Referenzen:

- „Datenquellenspezifische Einschränkungen für gespeicherte Abfragetabellen“ auf Seite 201

Erstellen einer gespeicherten Abfragetabelle in Systemen zusammengesetzter Datenbanken

Sie verwenden gespeicherte Abfragetabellen, um Daten lokal im Cache zu speichern und um die Leistung Ihrer Abfragen zu verbessern. Sie können Kurznamen relationaler und nicht relationaler Datenquellen verwenden, um gespeicherte Abfragetabellen zu erstellen.

Einschränkungen:

- Datenquellenspezifische Einschränkungen für gespeicherte Abfragetabellen.
- Wenn eine Abfrage über eine Funktionsschablone in einem Vergleichselement oder einer Auswahlliste verfügt, muss die Funktionsschablone Teil der gespeicherten Abfragetabelle sein.

Vorgehensweise:

Geben Sie zur Erstellung einer gespeicherten Abfragetabelle eine Anweisung CREATE TABLE ein, in der der Kurzname enthalten ist, der Ihr fernes Datenquellenobjekt darstellt.

Zugehörige Konzepte:

- „Gespeicherte Abfragetabellen und Systeme zusammenschlossener Datenbanken - Übersicht“ auf Seite 199

Zugehörige Tasks:

- „Erstellen einer gespeicherten Abfragetabelle“ in *Systemverwaltung: Implementierung*
- „Hinzufügen einer gespeicherten Abfragetabelle zu einer Cachetabelle“ auf Seite 209

Zugehörige Referenzen:

- „Datenquellenspezifische Einschränkungen für gespeicherte Abfragetabellen“ auf Seite 201
- „ALTER TABLE statement“ in *SQL Reference, Volume 2*
- „CREATE TABLE statement“ in *SQL Reference, Volume 2*

Datenquellenspezifische Einschränkungen für gespeicherte Abfragetabellen

In diesem Abschnitt werden die Einschränkungen bei der Erstellung gespeicherter Abfragetabellen beschrieben, die bei den folgenden Datenquellen gelten:

- Bio-RS
- BLAST
- Entrez
- HMMER
- IBM Lotus Extended Search
- Dateien mit Tabellenstruktur
- Web-Services
- XML

Einschränkungen für Bio-RS, Entrez und IBM Lotus Extended Search

Für diese Wrapper ist mindestens ein Vergleichselement in der WHERE-Klausel erforderlich. Sie müssen eine gespeicherte Abfragetabelle erstellen, die den Vergleichselementanforderungen der Wrapper gerecht wird. Wenn Sie kein Vergleichselement angeben, schlägt die Aktualisierung der gespeicherten Abfragetabelle fehl.

Einschränkungen für BLAST und HMMER

Wenn Sie einen BLAST- oder HMMER-Wrapper verwenden, fordert die Datenquelle für einige Vergleichselemente Werte an. Der Wrapper stellt den Standardwert zur Verfügung. Für andere Vergleichselemente müssen Sie einen Wert angeben. Wenn Sie kein Vergleichselement angeben, schlägt die Aktualisierung der gespeicherten Abfragetabelle fehl.

Im folgenden Beispiel wird die gespeicherte Abfragetabelle erfolgreich erstellt, da der Wrapper nicht auf die Datenquelle zugegriffen hat.

```
CREATE TABLE MY_MQT AS (SELECT * FROM BLAST_NICK)
DATA INITIALLY DEFERRED REFRESH DEFERRED ENABLE QUERY OPTIMIZATION;
```

Wenn Sie aber die Tabelle aktualisieren, wird auf die BLAST-Datenquelle zugegriffen, um Daten abzurufen. Die BLAST-Datenquelle gibt einen Fehler aus, da in der Spalte blast_seq ein Vergleichselement erforderlich ist.

Wenn Sie bei optionalen Vergleichselementen eine Abfrage eingeben und hierbei nicht die Standardwerte benutzen, müssen Sie beim Erstellen der gespeicherten Abfragetabelle ebenfalls Werte angeben. Wenn Sie kein Vergleichselement angeben, schlägt die Aktualisierung der gespeicherten Abfragetabelle fehl.

Im folgenden Beispiel stellt der Wrapper die Standardwerte für die optionalen Parameter zur Verfügung, wenn Sie die gespeicherte Abfragetabelle erstellen. Diese optionalen Parameter lauten in_arg1 = 2 und in_arg2 < 30.

```
CREATE TABLE MY_MQT AS (SELECT * FROM BLAST_NICK WHERE BLAST_SEQ = '12345')
  DATA INITIALLY DEFERRED REFRESH DEFERRED ENABLE QUERY OPTIMIZATION;
```

Wenn Sie die folgende Abfrage eingeben, stimmt das Optimierungsprogramm die Abfrage auf die gespeicherte Abfragetabelle ab. Wenn das Optimierungsprogramm den Plan der gespeicherten Abfragetabelle auswählt, ergeben sich aus dieser Abfrage falsche Ergebnisse. Die Abfrage gibt falsche Ergebnisse zurück, da die gespeicherte Abfragetabelle Daten mit in_arg1 = 2 enthält, die Abfrage aber Daten mit in_arg1 = 3 angefordert hat.

```
SELECT * FROM BLAST_NICK WHERE BLAST_SEQ = '12345' AND IN_ARG1 = 3;
```

Wenn Sie eine gespeicherte Abfragetabelle erstellen, müssen Sie die Werte für einige Vergleichselemente explizit angeben, auch wenn der Wrapper Standardwerte zur Verfügung stellt. Andernfalls kann das Optimierungsprogramm möglicherweise die Abfrage nicht an eine übereinstimmende gespeicherte Abfragetabelle weiterleiten. Zu dieser Situation kann es kommen, weil Sie diese Vergleichselementwerte bei der Erstellung der gespeicherten Abfragetabelle nicht explizit angegeben haben.

Im folgenden Beispiel sehen Sie, wie das Optimierungsprogramm bei der Auswahl der gespeicherten Abfragetabelle fehlschlägt, wenn die Abfrage mit der gespeicherten Abfragetabelle übereinstimmt. Wenn die gespeicherte Abfragetabelle ausdrücklich eine festgelegte Eingabespalte mit einem Standardwert enthält und die Abfrage über dieses Vergleichselement nicht verfügt, wird die Abfrage nicht an die gespeicherte Abfragetabelle weitergeleitet.

```
CREATE TABLE K55ADMIN.BLAST_NICK1_M1 AS (
  SELECT SCORE, E_VALUE, QUERYSTRANDS
  FROM K55ADMIN.BLAST_NICK1
  WHERE BLASTSEQ='ATGATCGGATCGAATTCGAT'
  AND E_VALUE < 10) DATA INITIALLY DEFERRED REFRESH DEFERRED;
```

Wenn Sie die folgende Abfrage eingeben, wird die Abfrage nicht an die gespeicherte Abfragetabelle weitergeleitet. Die Abfrage wird nicht weitergeleitet, weil das Optimierungsprogramm feststellt, dass die gespeicherte Abfragetabelle nicht E_VALUE < 10 angegeben hat.

```
SELECT SCORE, E_VALUE, QUERYSTRANDS
FROM K55ADMIN.BLAST_NICK1 WHERE BLASTSEQ='ATGATCGGATCGAATTCGAT';
```

Einschränkungen für Dateien mit Tabellenstruktur

Wenn Sie mit der Option DOCUMENT einen Kurznamen für eine Datei mit Tabellenstruktur definieren, muss die gespeicherte Abfragetabelle über ein Vergleichselement verfügen, das den Dateipfad angibt. Wenn Sie kein Vergleichselement angeben, schlägt die Aktualisierung der gespeicherten Abfragetabelle fehl.

Einschränkungen für Web-Services

Sie können eine gespeicherte Abfragetabelle nur über eine Übersichtsanzeige einer Hierarchie von Kurznamen erstellen. Sie können nicht für jeden Kurznamen einer Hierarchie eine gespeicherte Abfragetabelle erstellen.

Einschränkungen für XML

Sie können keine gespeicherte Tabelle für eine untergeordnete Tabelle erstellen.

Wenn Sie mit der Option DOCUMENT einen Kurznamen für eine XML-Tabelle definieren, fordert die gespeicherte Abfragetabelle ein Vergleichselement an, das den Dateipfad angibt. Wenn Sie kein Vergleichselement angeben, schlägt die Aktualisierung der gespeicherten Abfragetabelle fehl.

Zugehörige Tasks:

- „Erstellen einer gespeicherten Abfragetabelle in Systemen zusammengesetzter Datenbanken“ auf Seite 200

Zugehörige Referenzen:

- „Einschränkungen bei der Verwendung gespeicherter Abfragetabellen mit Kurznamen“ auf Seite 204

Einschränkungen bei der Verwendung gespeicherter Abfragetabellen mit Kurznamen

DB2 Information Integrator unterstützt keine vom System verwalteten gespeicherten Abfragetabellen, die auf Kurznamen in einer partitionierten Datenbankumgebung verweisen.

Um diese Einschränkung zu umgehen, können Sie vom Benutzer verwaltete gespeicherte Abfragetabellen verwenden.

Sie können z. B. für den nicht relationalen Kurznamen DEPART die folgenden Befehle zur Simulation einer vom System verwalteten gespeicherten Abfragetabelle eingeben.

```
SET CURRENT MAINTAINED TABLE TYPES FOR OPTIMIZATION ALL;  
  
CREATE TABLE AST1(C1, C2)  
  AS (SELECT EMPNO, FIRSTNME FROM DEPART WHERE EMPNO>'000000')  
  DATA INITIALLY DEFERRED REFRESH DEFERRED  
  ENABLE QUERY OPTIMIZATION MAINTAINED BY USER;  
  
SET INTEGRITY FOR AST1 ALL IMMEDIATE UNCHECKED;  
  
INSERT INTO AST1 (SELECT EMPNO, FIRSTNME FROM DEPART WHERE EMPNO>'000000');  
  
SET CURRENT REFRESH AGE ANY;
```

Die folgende Anweisung SELECT kann von der zuvor definierten gespeicherten Abfragetabelle beantwortet werden:

```
SELECT EMPNO, FIRSTNME FROM DEPART  
  WHERE EMPNO > '000000' AND FIRSTNME LIKE 'AN%';
```

Kapitel 14. Cachetabellen in einem System zusammengesetzter Datenbanken

In diesem Kapitel wird beschrieben, was eine Cachetabelle ist und wie eine solche Tabelle in Ihrem System zusammengesetzter Datenbanken zum Einsatz kommt.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Cachetabellen“
- „Erstellen einer Cachetabelle“ auf Seite 207
- „Aktivieren eines Caches“ auf Seite 208
- „Hinzufügen einer gespeicherten Abfragetabelle zu einer Cachetabelle“ auf Seite 209
- „Löschen einer gespeicherten Abfragetabelle aus einer Cachetabelle“ auf Seite 210
- „Löschen einer Cachetabelle“ auf Seite 210

Cachetabellen

Durch eine Cachetabelle kann die Abfrageleistung verbessert werden, da in diesem Fall auf eine lokal gespeicherte Untergruppe der Daten und nicht direkt auf die Daten zugegriffen wird, die in der fernen relationalen Datenquelle gespeichert sind. Eine Cachetabelle besteht aus den folgenden Komponenten:

- Einem Kurznamen auf Ihrem System zusammengesetzter Datenbanken mit denselben Spaltendefinitionen und denselben Datenzugriffsmöglichkeiten wie für die Tabelle mit der fernen relationalen Datenquelle.
- Einer oder mehreren vom Benutzer verwalteten gespeicherten Abfragetabellen, die Sie für den Kurznamen definieren. Der Kurzname kann eine Untergruppe mit häufig verwendeten Daten aus Ihrer fernen Datenquelle enthalten.
- Einem benutzerdefinierten Replikationsplan, der jeder gespeicherten Abfragetabelle zugeordnet ist, um die lokalen gespeicherten Abfragetabellen auf dem aktuellen Stand zu halten wie die ferne Datenquelle.

In Abb. 9 auf Seite 206 sehen Sie das Konzept einer Cachetabelle.

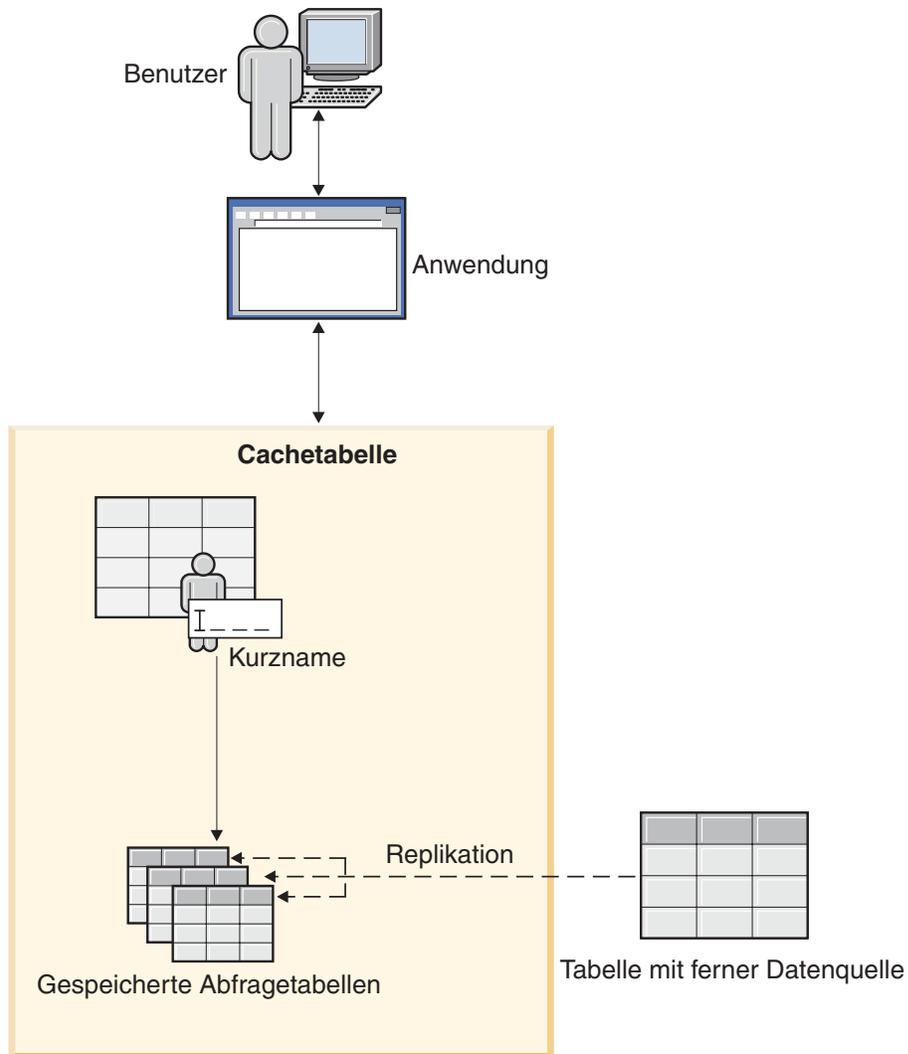


Abbildung 9. Cachetabellen bestehen aus einem Kurznamen, gespeicherten Abfragetabellen und einem Replikationsplan.

Die Cachetabelle hat denselben Namen wie die Kurznamenkomponente. Sie können eine Cachetabelle nur einer einzigen fernen Tabelle zuordnen. Die Cachetabelle kann eine vollständige Kopie oder eine Teilmenge von Zeilen aus Ihrer fernen Datenquelle umfassen. Die Cachetabelle enthält lokale Daten, die durch die zugehörigen gespeicherten Abfragetabellen definiert wurden. Der Hauptunterschied zwischen einer Cachetabelle und einer gespeicherten Abfragetabelle besteht darin, dass eine Cachetabelle von der DB2®-Steuerzentrale aus konfiguriert wird.

Anwendungen, mit denen eine ferne Datenquelle abgefragt wird, können mit minimalem Aufwand so geändert werden, dass Abfragen in der Cachetabelle ausgeführt werden können. Bei der Abfrageverarbeitung leitet das Optimierungsprogramm die Abfrage an die Cachetabelle oder die Tabelle mit der fernen relationalen Datenquelle weiter.

Zugehörige Konzepte:

- „Replizierte gespeicherte Abfragetabellen“ in *Systemverwaltung: Konzept*
- „Gespeicherte Abfragetabellen und Systeme zusammengeschnittener Datenbanken - Übersicht“ auf Seite 199

Zugehörige Tasks:

- „Erstellen einer Cachetabelle“ auf Seite 207
- „Erstellen einer gespeicherten Abfragetabelle in Systemen zusammengeschnittener Datenbanken“ auf Seite 200

Erstellen einer Cachetabelle

Verwenden Sie eine Cachetabelle zur Speicherung von Daten, auf die Sie häufig zugreifen, die jedoch selten geändert werden. Durch die Verwendung von Cachetabellen können Sie die Abfrageleistung verbessern.

Sie können Daten aus Datenquellen im Cache speichern, für die unter DB2 UDB eine heterogene Replikation unterstützt wird. Diese Datenquellen sind Datenquellen der DB2-Produktfamilie sowie Informix-, Microsoft SQL Server-, Oracle- und Sybase-Datenquellen.

Voraussetzungen:

- Setzen Sie den Konfigurationsparameter FEDERATED des Datenbankmanagers für den DB2 UDB-Server auf YES (Ja).
- Installieren und konfigurieren Sie die Software für Informix Client SDK auf dem Server mit zusammengeschnittenen Datenbanken, um auf die Informix-Datenquellen zuzugreifen.
- Um Daten einer Tabelle von DB2 UDB für Linux, UNIX und Windows im Cache zu speichern, müssen Sie die Datenbank, in der die Tabelle gespeichert ist, für die *Protokollierung mit Protokollspeicherung* konfigurieren. Die Protokollierung mit Protokollspeicherung ist eine Art Archivprotokollierung. Setzen Sie den Wert für LOGRETAIN auf RECOVER, um die Datenbank für die Protokollierung mit Protokollspeicherung zu konfigurieren.
- Die zusammengeschnittene Datenbank bzw. die Quelldatenbank muss sich auf dem Computer befinden, von dem aus Sie die Cachetabellen erstellen. Wenn die zusammengeschnittene Datenbank bzw. die Quelldatenbank nicht lokal gespeichert ist, müssen Sie die Datenbanken auf dem lokalen Computer katalogisieren. Der Aliasname, den Sie beim Katalogisieren der Datenbank verwenden, muss mit dem Datenbanknamen identisch sein.
- Die Benutzer-ID, die für die Benutzerzuordnung zwischen den Datenbanken definiert wurde, muss über die Berechtigung zum Erstellen von Tabellen in der Quelldatenbank verfügen.

Vorgehensweise:

Gehen Sie wie folgt vor, um eine Cachetabelle zu erstellen:

1. Erweitern Sie in der DB2-Steuerzentrale den Ordner **Cacheobjekte**.
2. Klicken Sie mit der rechten Maustaste auf den Ordner **Cachetabellen**, und klicken Sie dann auf **Erstellen**. Der Cachetabellenassistent wird geöffnet.
3. Führen Sie die Schritte im Assistenten aus.

Der Cachetabellenassistent erstellt eine einzige gespeicherte Abfragetabelle, wenn Sie die Cachetabelle erstellen. Sie können zusätzliche gespeicherte Abfragetabellen erstellen, um weitere Daten aus derselben Datenquelle zu speichern.

Zugehörige Konzepte:

- „Cachetabellen“ auf Seite 205

Zugehörige Tasks:

- „Aktivieren eines Caches“ auf Seite 208
- „Hinzufügen einer gespeicherten Abfragetabelle zu einer Cachetabelle“ auf Seite 209

Zugehörige Referenzen:

- „logretain - Beibehalten von Protokollen aktivieren“ in *Systemverwaltung: Optimierung*

Aktivieren eines Caches

Wenn Sie einen Cache aktivieren, aktivieren Sie die Capture- und Apply-Programme. Diese Programme replizieren Daten aus der Datenquelle in die gespeicherte Abfragetabelle.

Wenn Sie eine Cachetabelle erstellen, wird normalerweise der Cache aktiviert. In folgenden Fällen kann eine manuelle Aktivierung eines Caches für die Replikation erforderlich sein:

- Wenn Sie eine Cachetabelle erstellen und die beiden folgenden Bedingungen zutreffen, hat die Datenbank den Cache nicht für die Replikation aktiviert:
 - Die Datenquelle ist eine Tabelle von DB2 UDB für Linux, UNIX und Windows.
 - Sie haben die Datenbank nicht konfiguriert, in der sich die Tabelle für die Protokollierung mit Protokollspeicherung befindet.
- Nach der Erstellung einer Cachetabelle wird ein Cache inaktiviert, wenn das Capture- bzw. Apply-Programm für die Replikation gestoppt wird. Wenn Sie das Capture- bzw. Apply-Programm in der DB2-Replikationszentrale erneut starten, müssen Sie den Cache aktivieren, damit die Replikation der Daten gestartet wird.

In der DB2-Steuerzentrale wird für die *Protokollierung mit Protokollspeicherung* gelegentlich der Terminus *Archivprotokollierung* verwendet.

Voraussetzungen:

Wenn der Cache für eine Tabelle von DB2 UDB für Linux, UNIX und Windows definiert wurde, gelten die folgenden Voraussetzungen:

- Sie müssen die Datenbank konfigurieren, in der sich die Tabelle für die Protokollierung mit Protokollspeicherung befindet.
- Der Datenbankverwaltungsserver (DAS = Database Administration Server) muss ausgeführt werden.

Vorgehensweise:

Gehen Sie wie folgt vor, um einen Cache zu aktivieren:

1. Erweitern Sie in der DB2-Steuerzentrale in der Objektbaumstruktur den Ordner **Cacheobjekte** und den Ordner **Cachetabellen**.

2. Klicken Sie mit der rechten Maustaste auf die Cachetabelle, in der sich der Cache befindet, den Sie aktivieren möchten, und klicken Sie auf **Caching aktivieren**. Das Fenster **Caching aktivieren** wird geöffnet.
3. Wählen Sie den Cache aus, den Sie aktivieren möchten, und klicken Sie auf **OK**.

Zugehörige Konzepte:

- „Cachetabellen“ auf Seite 205

Zugehörige Tasks:

- „Hinzufügen einer gespeicherten Abfragetabelle zu einer Cachetabelle“ auf Seite 209

Hinzufügen einer gespeicherten Abfragetabelle zu einer Cachetabelle

Wenn Sie eine Cachetabelle erstellen, speichert der Server mit zusammengeschlossenen Datenbanken die Daten separat von der Datenquelle in einer gespeicherten Abfragetabelle auf der lokalen Einheit. Die Kriterien, die Sie im Cachetabellenassistenten angeben, bestimmen, welche Daten in der gespeicherten Abfragetabelle gespeichert werden sollen.

Sie können zusätzliche gespeicherte Abfragetabellen für dieselbe Cachetabelle erstellen, um weitere Daten aus demselben Datenquellenobjekt zu speichern.

Voraussetzungen:

Für die Datenquelle muss eine Cachetabelle vorhanden sein.

Vorgehensweise:

Gehen Sie wie folgt vor, um einer vorhandenen Cachetabelle eine gespeicherte Abfragetabelle hinzuzufügen:

1. Erweitern Sie in der DB2-Steuerzentrale in der Objektbaumstruktur den Ordner **Cacheobjekte** und den Ordner **Cachetabellen**.
2. Klicken mit der rechten Maustaste auf die Cachetabelle, der Sie eine gespeicherte Abfragetabelle hinzufügen möchten, und klicken Sie auf **Merkmale**. Das Fenster **Merkmale der Cachetabelle** wird geöffnet.
3. Klicken Sie auf **Hinzufügen**. Der Cachetabellenassistent wird geöffnet.
4. Führen Sie die Schritte im Assistenten aus.

Zugehörige Konzepte:

- „Cachetabellen“ auf Seite 205
- „Gespeicherte Abfragetabellen und Systeme zusammengeschlossener Datenbanken - Übersicht“ auf Seite 199

Zugehörige Tasks:

- „Aktivieren eines Caches“ auf Seite 208
- „Löschen einer gespeicherten Abfragetabelle aus einer Cachetabelle“ auf Seite 210

Löschen einer gespeicherten Abfragetabelle aus einer Cachetabelle

Wenn Sie keine Daten mehr lokal in einer gespeicherten Abfragetabelle speichern möchten, können Sie die gespeicherte Abfragetabelle aus der Cachetabelle löschen. Wenn eine Cachetabelle nur über eine gespeicherte Abfragetabelle verfügt, wird durch das Löschen der gespeicherten Abfragetabelle auch die Cachetabelle gelöscht.

Einschränkungen:

Verwenden Sie die DB2-Steuerzentrale zum Löschen einer gespeicherten Abfragetabelle aus einer Cachetabelle, um sicherzustellen, dass die gesamte gespeicherte Abfragetabelle von Ihrem System entfernt wurde.

Vorgehensweise:

Gehen Sie wie folgt vor, um eine gespeicherte Abfragetabelle aus einer vorhandenen Cachetabelle zu löschen:

1. Erweitern Sie in der DB2-Steuerzentrale in der Objektbaumstruktur den Ordner **Cacheobjekte** und den Ordner **Cachetabellen**.
2. Klicken mit der rechten Maustaste auf die Cachetabelle, aus der Sie eine gespeicherte Abfragetabelle löschen möchten, und klicken Sie auf **Merkmale**. Das Fenster **Merkmale der Cachetabelle** wird geöffnet.
3. Wählen Sie die zu löschende gespeicherte Abfragetabelle aus, und klicken Sie auf **Entfernen**.

Zugehörige Tasks:

- „Hinzufügen einer gespeicherten Abfragetabelle zu einer Cachetabelle“ auf Seite 209
- „Löschen einer Cachetabelle“ auf Seite 210

Löschen einer Cachetabelle

Wenn Sie keine Daten mehr lokal in einer Cachetabelle speichern möchten, können Sie die Cachetabelle löschen.

Das Löschen einer Cachetabelle bewirkt die Ausführung folgender Aktionen:

- Die für die Cachetabelle erstellten gespeicherten Abfragetabellen werden gelöscht.
- Der Replikationsplan zwischen den Datenquellen und den gespeicherten Abfragetabellen wird entfernt.
- Der Kurzname für die Datenquelle wird gelöscht, wenn der Kurzname bei der Erstellung der Cachetabelle erstellt wurde. Wenn Sie bei der Erstellung der Cachetabelle einen vorhandenen Kurznamen verwendet haben, löscht der Server mit zusammengeschlossenen Datenbanken den Kurznamen nicht.

| **Vorgehensweise:**

| Gehen Sie wie folgt vor, um eine Cachetabelle zu löschen:

- | 1. Erweitern Sie in der DB2-Steuerzentrale in der Objektbaumstruktur den Ordner
| **Cacheobjekte** und den Ordner **Cachetabellen**.
- | 2. Klicken Sie mit der rechten Maustaste auf die zu löschende Cachetabelle, und
| klicken Sie auf **Löschen**.

| **Zugehörige Tasks:**

- | • „Erstellen einer Cachetabelle“ auf Seite 207
- | • „Löschen einer gespeicherten Abfragetabelle aus einer Cachetabelle“ auf Seite
| 210

Kapitel 15. Informative Integritätsbedingungen für Kurznamen in einem System zusammengesetzter Datenbanken

In diesem Kapitel wird die Verwendung der informativen Integritätsbedingungen für Kurznamen beschrieben.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Informative Integritätsbedingungen für Kurznamen“
- „Angaben von informativen Integritätsbedingungen für Kurznamen“
- „Angaben von informativen Integritätsbedingungen für Kurznamen - Beispiele“ auf Seite 215

Informative Integritätsbedingungen für Kurznamen

Informative Integritätsbedingungen sind Regeln, die das Optimierungsprogramm zur Leistungsverbesserung verwenden kann, die vom Datenbankmanager jedoch nicht zwingend angewendet werden. Sie können informative Integritätsbedingungen für Kurznamen verwenden, um die Leistung von Abfragen ferner Datenquellen zu verbessern.

Sie können die folgenden Typen informativer Integritätsbedingungen für Kurznamen angeben:

- Referenzielle Integritätsbedingungen
- Prüfung auf Integritätsbedingungen
- Integritätsbedingungen der funktionalen Abhängigkeiten
- Integritätsbedingungen für Primärschlüssel
- Eindeutige Integritätsbedingungen

Zugehörige Konzepte:

- „Constraints“ in *SQL Reference, Volume 1*

Zugehörige Tasks:

- „Angaben von informativen Integritätsbedingungen für Kurznamen“ auf Seite 213

Zugehörige Referenzen:

- „Angaben von informativen Integritätsbedingungen für Kurznamen - Beispiele“ auf Seite 215

Angaben von informativen Integritätsbedingungen für Kurznamen

Um die Leistung einiger Abfragen für ferne Datenquellen zu verbessern, können Sie informative Integritätsbedingungen für Kurznamen hinzufügen. Allerdings erzwingt oder überprüft der Server mit zusammengesetzten Datenbanken die Integritätsbedingungen nicht.

Für relationale Datenquellen können Sie informative Integritätsbedingungen bei der Änderung eines Kurznamens angeben.

Für nicht relationale Datenquellen können Sie diese bei der Erstellung oder Änderung eines Kurznamens definieren.

Einschränkungen:

Wenn Sie informative Integritätsbedingungen zu einem Kurznamen definiert haben, können Sie die Spaltennamen für diesen Kurznamen erst wieder ändern, nachdem Sie die informativen Integritätsbedingungen entfernt haben.

Vorgehensweise:

Von der DB2-Steuerzentrale oder der DB2-Befehlszeile aus können Sie informative Integritätsbedingungen zu einem Kurznamen angeben.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Wählen Sie den Ordner **Kurznamen** aus:

- Wenn Sie einen Kurznamen erstellen, klicken Sie im Teilfenster mit den Objektdetails der DB2-Steuerzentrale in der Liste der Aktionen auf **Kurznamen erstellen**. Der Kurznamenassistent wird geöffnet.
Öffnen Sie im Fenster **Kurznamen erstellen** das Notizbuch **Kurznamen hinzufügen** oder das Notizbuch **Merkmale** für einen Kurznamen:
 - Wenn Sie nur einen einzelnen Kurznamen erstellen möchten, klicken Sie auf **Hinzufügen**. Das Notizbuch **Kurznamen hinzufügen** wird geöffnet.
 - Wenn Sie zur Generierung einer Liste mit Kurznamen die Aufspürfunktion verwendet haben, wählen Sie den Kurznamen aus, dem Sie informative Integritätsbedingungen hinzufügen möchten. Klicken Sie anschließend auf **Merkmale**. Das Notizbuch **Merkmale** wird geöffnet.
- Wenn Sie einen Kurznamen ändern, klicken Sie auf den zu ändernden Kurznamen. Klicken Sie im Teilfenster mit den Objektdetails der DB2-Steuerzentrale in der Liste der Aktionen auf **Ändern**. Das Notizbuch **Kurznamen ändern** wird geöffnet.

2. Definieren Sie auf der Seite **Schlüssel** die referenziellen Integritätsbedingungen für den Kurznamen. Sie können eine über Primärschlüssel, eindeutige Schlüssel oder Fremdschlüssel definierte Integritätsbedingung festlegen.

3. Legen Sie auf der Seite **Prüfung auf Integritätsbedingung** die Prüfungen auf Integritätsbedingung bzw. die Integritätsbedingungen der funktionalen Abhängigkeiten für den Kurznamen fest.

4. Klicken Sie auf **OK**, um die informativen Integritätsbedingungen festzulegen, und schließen Sie das Notizbuch.

Gehen Sie wie folgt vor, um diese Task von der DB2-Befehlszeile aus auszuführen:

Geben Sie die Anweisung CREATE NICKNAME bzw. ALTER NICKNAME mit den entsprechenden festgelegten Integritätsbedingungsattributen ein.

Zugehörige Konzepte:

- „Primärschlüssel“ in *Systemverwaltung: Konzept*
- „Informative Integritätsbedingungen für Kurznamen“ auf Seite 213

Zugehörige Referenzen:

- „Angaben von informativen Integritätsbedingungen für Kurznamen - Beispiele“ auf Seite 215

Angeben von informativen Integritätsbedingungen für Kurznamen - Beispiele

Die folgenden Beispiele veranschaulichen die Verwendung informativer Integritätsbedingungen für Kurznamen.

Informative Prüfung auf Integritätsbedingung

In der folgenden fernen Tabelle sind die Daten in der Spalte 'salary' immer größer als 10000.

```
CREATE TABLE account.salary (  
    empno INTEGER NOT NULL PRIMARY KEY,  
    salary INTEGER NOT NULL  
);
```

Erstellen Sie für diese Tabelle einen Kurznamen:

```
CREATE NICKNAME account.salary FOR myserv.account.salary;
```

Fügen Sie dann die informativen Prüfungen auf Integritätsbedingung für den Kurznamen hinzu, indem Sie die folgende Anweisung eingeben:

```
ALTER NICKNAME account.salary ADD CONSTRAINT cons1 CHECK( salary > 10000 )  
NOT ENFORCED  
ENABLE QUERY OPTIMIZATION;
```

Informative referenzielle Integritätsbedingung: Kurzname zu Kurzname

In diesem Beispiel finden Sie die beiden Kurznamen N1 und N2. Die Spalte F1 des Kurznamens N2 enthält den Schlüsselwert in Spalte P1 des Kurznamens N1. Sie können die referenzielle Integritätsbedingung für den Kurznamen N2 definieren, indem Sie die folgende Anweisung eingeben:

```
ALTER NICKNAME SCHEMA1.N2 ADD CONSTRAINT ref1  
    FOREIGN KEY (F1) REFERENCES SCHEMA1.N1 (P1)  
    NOT ENFORCED;
```

Informative referenzielle Integritätsbedingung: Kurzname zu Tabelle

In diesem Beispiel enthält Kurzname N3 mit Spalte F1 den Schlüsselwert in Spalte P1 von Tabelle T1. Sie können die referenzielle Integritätsbedingung für den Kurznamen N3 definieren, indem Sie die folgende Anweisung eingeben:

```
ALTER NICKNAME SCHEMA1.N3 ADD CONSTRAINT ref1  
    FOREIGN KEY (F1) REFERENCES SCHEMA1.T1 (P1)  
    NOT ENFORCED;
```

Informative referenzielle Integritätsbedingung: Tabelle zu Kurzname

In diesem Beispiel enthält Tabelle T2 mit Spalte F1 den Schlüsselwert in Spalte P1 von Kurzname N4. Sie können die referenzielle Integritätsbedingung für Tabelle T2 definieren, indem Sie die folgende Anweisung eingeben:

```
ALTER TABLE SCHEMA1.T2 ADD CONSTRAINT ref1  
    FOREIGN KEY (F1) REFERENCES SCHEMA1.N4 (P1)  
    NOT ENFORCED;
```

Funktionale Abhängigkeit

In diesem Beispiel legt das Spaltenpaar C1 und C2 eindeutig den Wert in Spalte P1 fest. Sie können eine funktionale Abhängigkeit definieren, indem Sie die folgende Anweisung eingeben:

```
ALTER NICKNAME SCHEMA1.NICK1 ADD CONSTRAINT FD1 CHECK( P1 DETERMINED BY (C1,C2) )
NOT ENFORCED ENABLE QUERY OPTIMIZATION;
```

Datei mit Tabellenstruktur

In diesem Beispiel sehen Sie, wie ein Primärschlüssel für eine Datei mit Tabellenstruktur definiert wird.

```
CREATE NICKNAME MY_FILE (
    X INTEGER NOT NULL,
    Y INTEGER,
    PRIMARY KEY (X) NOT ENFORCED
) FOR SERVER MY_SERVER OPTIONS(FILE_PATH '/usr/pat/DRUGDATA1.TXT');
```

Sternschema

Im folgenden Beispiel sehen Sie vier Dimensionstabellen und eine Fakttablelle.

```
CREATE TABLE SCHEMA.FACT (
    LOCATION_CODE INTEGER NOT NULL,
    PRODUCT_CODE INTEGER NOT NULL,
    CUSTOMER_CODE INTEGER NOT NULL,
    SDATE DATE NOT NULL,
    SALES INTEGER NOT NULL
);

CREATE TABLE SCHEMA.LOCATION (
    LOCATION_CODE INTEGER NOT NULL PRIMARY KEY,

    STATE CHAR(2) NOT NULL,
    SHOP_ID INTEGER NOT NULL,
    ...
);

CREATE TABLE SCHEMA.PRODUCT (
    PRODUCT_CODE INTEGER NOT NULL PRIMARY KEY,
    PRODUCT_CAT INTEGER NOT NULL,
    PRODUCT_NAME VARCHAR(20) NOT NULL,
    ...
);

CREATE TABLE SCHEMA.CUSTOMER (
    CUSTOMER_CODE INTEGER NOT NULL PRIMARY KEY,
    NAME VARCHAR(20) NOT NULL,
    TEL VARCHAR(10) NOT NULL,
    ...
);

CREATE TABLE SCHEMA.TIMEDIM (
    SDATE DATE NOT NULL UNIQUE,
    YEAR INTEGER NOT NULL,
    QUARTER INTEGER NOT NULL,
    ...
);
```

Der Server mit zusammengeschlossenen Datenbanken erstellt für die Fakttable und die vier Dimensionstabellen die folgenden Kurznamen:

```
CREATE NICKNAME SCHEMA.FACT FOR SERVER.SCHEMA.FACT;  
CREATE NICKNAME SCHEMA.LOCATION FOR SERVER.SCHEMA.LOCATION;  
CREATE NICKNAME SCHEMA.PRODUCT FOR SERVER.SCHEMA.PRODUCT;  
CREATE NICKNAME SCHEMA.CUSTOMER FOR SERVER.SCHEMA.CUSTOMER;  
CREATE NICKNAME SCHEMA.TIMEDIM FOR SERVER.SCHEMA.TIMEDIM;
```

Sie können die folgende Fremdschlüsselrelation definieren:

```
ALTER NICKNAME SCHEMA.FACT ADD CONSTRAINT L1 FOREIGN KEY (LOCATION_CODE)  
REFERENCES SCHEMA.LOCATION(LOCATION_CODE)  
NOT ENFORCED ENABLE QUERY OPTIMIZATION;  
  
ALTER NICKNAME SCHEMA.FACT ADD CONSTRAINT P1 FOREIGN KEY (PRODUCT_CODE)  
REFERENCES SCHEMA.PRODUCT(PRODUCT_CODE)  
NOT ENFORCED ENABLE QUERY OPTIMIZATION;  
  
ALTER NICKNAME SCHEMA.FACT ADD CONSTRAINT C1 FOREIGN KEY (CUSTOMER_CODE)  
REFERENCES SCHEMA.CUSTOMER(CUSTOMER_CODE)  
NOT ENFORCED ENABLE QUERY OPTIMIZATION;  
  
ALTER NICKNAME SCHEMA.FACT ADD CONSTRAINT S1 FOREIGN KEY (SDATE)  
REFERENCES SCHEMA.TIMEDIM(SDATE)  
NOT ENFORCED ENABLE QUERY OPTIMIZATION;
```

Wenn der Wert der Spalte TEL im Kurznamen CUSTOMER eindeutig ist, können Sie die folgende eindeutige informative Integritätsbedingung hinzufügen:

```
ALTER NICKNAME SCHEMA.CUSTOMER ADD CONSTRAINT U1 UNIQUE( TEL )  
NOT ENFORCED ENABLE QUERY OPTIMIZATION;
```

Wenn der Wert der Spalte SHOP_ID im Kurznamen LOCATION eindeutig den Wert der Spalte LOCATION_ID bestimmt, können Sie die folgende funktionale Abhängigkeit definieren:

```
ALTER NICKNAME SCHEMA.LOCATION  
ADD CONSTRAINT F1 CHECK( LOCATION_ID DETERMINED BY SHOP_ID )  
NOT ENFORCED ENABLE QUERY OPTIMIZATION;
```

Da der Wert in der Spalte QUARTER im Kurznamen TIMEDIM zwischen 1 und 4 liegt, können Sie die folgende informative Prüfung auf Integritätsbedingung definieren:

```
ALTER NICKNAME SCHEMA.TIMEDIM  
ADD CONSTRAINT Q1 CHECK( QUARTER BETWEEN 1 AND 4 )  
NOT ENFORCED ENABLE QUERY OPTIMIZATION;
```

Mit den Anweisungen in diesem Beispiel werden Kurznamen für ferne Tabellen erstellt. Kurznamen verfügen über Primärschlüssel, wenn die fernen Tabellen über Primärschlüssel verfügen. Wenn Sie für Sichten Kurznamen erstellen, verfügen die Kurznamen über keine Primärschlüssel. In diesem Fall können Sie den Kurznamen ändern, um eine informative, über Primärschlüssel definierte Integritätsbedingung hinzuzufügen. Beispiel:

```
CREATE NICKNAME SCHEMA.LOCATION FOR SERVER.SH.V_LOCATION;  
ALTER NICKNAME SCHEMA.LOCATION  
ADD CONSTRAINT P1 PRIMARY KEY ( LOCATION_CODE ) NOT ENFORCED;
```

Zugehörige Tasks:

- „Angabe von informativen Integritätsbedingungen für Kurznamen“ auf Seite 213

Kapitel 16. Kurznamenstatistiken

In diesem Kapitel wird beschrieben, wie Kurznamenstatistiken abgerufen werden.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht“
- „Abrufen von Kurznamenstatistiken“ auf Seite 220
- „Abrufen von Kurznamenstatistiken von der Befehlszeile aus - Beispiele“ auf Seite 222
- „Erstellen eines DB2-Toolskatalogs“ auf Seite 223
- „Anzeigen des Aktualisierungsstatus der Kurznamenstatistiken“ auf Seite 224

Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht

Wenn Sie für ein Datenquellenobjekt einen Kurznamen registrieren, fügt der Server mit zusammengeschlossenen Datenbanken dem Systemkatalog in der zusammengeschlossenen Datenbank Informationen über dieses Datenquellenobjekt hinzu. Das DB2® UDB-Abfrageoptimierungsprogramm verwendet diese Informationen, um das Abrufen von Daten aus dem Datenquellenobjekt zu planen. Die zusammengeschlossene Datenbank stellt Änderungen an den Datenquellenobjekten nicht automatisch fest. Die Informationen im Systemkatalog sind daher möglicherweise nicht immer auf dem aktuellen Stand. Das Abrufen der Statistikdaten für einen Kurznamen stellt sicher, dass das Abfrageoptimierungsprogramm bei der Generierung der Abfragezugriffspläne die aktuellsten Informationen zum betreffenden Kurznamen verwendet.

Sie können die momentan verfügbaren Statistikdaten zu Datenbankkurznamen, Spalten und Indizes von der fernen Datenquelle abrufen.

Sie können die Statistikdaten eines einzelnen Kurznamens oder aller Kurznamen in einem DB2-Schema einer bestimmten DB2-Serverdefinition abrufen. Wenn ein Teil der Abrufaktion fehlschlägt, macht die Datenbank die Änderungen mit einer ROLLBACK-Operation rückgängig.

Für relationale Datenquellen und ODBC-Datenquellen, bei denen die Statistikdaten nicht auf ihre Standardwerte gesetzt sind, können Sie die folgenden Statistikwerte abrufen, sofern diese in der fernen Quelle zur Verfügung stehen:

- CARD
- FPAGES
- NPAGES
- OVERFLOW
- COLCARD
- HIGH2KEY
- LOW2KEY
- NLEAF
- NLEVELS
- CLUSTERFACTOR
- CLUSTERRATIO

- FULLKEYCARD
- FIRSTKEYCARD

Für nicht relationale Datenquellen und ODBC-Datenquellen, bei denen die Statistikdaten auf ihre Standardwerte gesetzt sind, können Sie die folgenden Statistikwerte abrufen, sofern diese in der fernen Quelle zur Verfügung stehen:

- CARD
- COLCARD
- HIGH2KEY
- LOW2KEY
- FULLKEYCARD
- FIRSTKEYCARD

Sie können Kurznamenstatistiken unter Verwendung der DB2-Steuerzentrale oder des DB2-Befehlszeilenprozessors abrufen.

Für die folgenden Datenquellen können Sie Kurznamenstatistiken abrufen:

- BioRS
- DB2-Produktfamilie (DRDA)
- Documentum (außer Pseudospalten)
- Informix[®]
- Microsoft[®] SQL Server
- ODBC
- Oracle (NET8)
- Sybase (CTLIB)
- Dateien mit Tabellenstruktur
- Teradata
- XML (Rootkurzname)

Sie können Kurznamenstatistiken auf Servern mit zusammengeschlossenen Datenbanken abrufen, die die Betriebssysteme Microsoft Windows[®] 2000 und Windows NT[®], AIX[®], Solaris, HP-UX sowie Linux verwenden.

Zugehörige Tasks:

- „Abrufen von Kurznamenstatistiken“ auf Seite 220
- „Anzeigen des Aktualisierungsstatus der Kurznamenstatistiken“ auf Seite 224

Zugehörige Referenzen:

- „Abrufen von Kurznamenstatistiken von der Befehlszeile aus - Beispiele“ auf Seite 222
- Kapitel 32, „Gespeicherte Prozedur SYSPROC.NNSTAT“, auf Seite 341

Abrufen von Kurznamenstatistiken

Das Abrufen der Statistikdaten für einen Kurznamen stellt sicher, dass das Abfrageoptimierungsprogramm die momentan in der Datenquelle verfügbaren Informationen zu dem betreffenden Kurznamen verwendet. Sie können die Abfrageleistung verbessern, indem Sie die Statistikinformationen zu einem Kurz-

namen immer auf dem aktuellsten Stand halten. Allerdings sind die Statistikdaten immer nur so genau und aktuell wie die sich momentan in der fernen Quelle befindlichen Informationen.

Voraussetzungen:

Wenn Sie die Eingabeaufforderung der Befehlszeile zum Aktualisieren der Statistikdaten verwenden, gelten die folgenden Voraussetzungen:

- Der Server mit zusammengeschlossenen Datenbanken erstellt auf dem Server die Protokolldatei. Die Verzeichnisse, die Sie im Pfad auflisten, müssen vorhanden sein.
- Verwenden Sie bei Windows zwei Backslashes, um den Protokollpfad anzugeben. Beispiel: c:\\temp\\nnstat.log.

Einschränkungen:

Wenn sich der lokale Spaltenname oder -typ vom fernen Spaltennamen oder -typ unterscheidet, ruft das Aktualisierungsdienstprogramm für Kurznamenstatistiken keine Spaltenstatistiken ab.

Die Benutzer-ID, die Sie zur Herstellung einer Verbindung zur zusammengeschlossenen Datenbank verwenden, muss der Tabelle der fernen Datenquellen zugeordnet sein.

Vorgehensweise:

Sie können Kurznamenstatistiken von der DB2-Steuerzentrale oder der DB2-Befehlszeile aus aktualisieren.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Wählen Sie die Kurznamen aus, für die Sie aktuelle Statistikdaten abrufen möchten:
 - Führen Sie folgende Schritte aus, um die Statistiken für alle Kurznamen zu aktualisieren, die einer Serverdefinition zugeordnet wurden:
 - a. Erweitern Sie den Ordner **Objekte zusammengeschlossener Datenbanken**.
 - b. Erweitern Sie den Ordner **Serverdefinitionen** mit den zu aktualisierenden Kurznamen.
 - c. Klicken Sie mit der rechten Maustaste auf den Namen der Serverdefinition, und wählen Sie **Statistik** aus.
 - d. Wählen Sie **Aktualisieren** aus. Das Notizbuch **Statistik aktualisieren** wird geöffnet.
 - Erweitern Sie den Ordner **Kurznamen**, um die Statistiken für einen einzelnen Kurznamen zu aktualisieren.
 - a. Klicken Sie mit der rechten Maustaste auf den Kurznamen, den Sie aktualisieren möchten, und wählen Sie **Statistik** aus.
 - b. Wählen Sie **Aktualisieren** aus. Das Fenster **Statistik aktualisieren** wird geöffnet.
2. Wenn kein DB2-Toolskatalog vorhanden ist, wird ein Fenster angezeigt, von dem aus Sie den DB2-Toolskatalog erstellen können.

3. Geben Sie die Einstellungen für die Aktualisierung an:
 - Führen Sie folgende Schritte aus, um die Statistiken für alle Kurznamen zu aktualisieren, die einer Serverdefinition zugeordnet wurden:
 - a. Führen Sie auf der Seite **Kurznamen** die folgenden Schritte aus:
 - Wählen Sie das Schema aus, das den Kurznamen zugeordnet ist, deren Statistiken Sie aktualisieren möchten. Wenn Sie kein Schema auswählen, aktualisiert der Server mit zusammengeschlossenen Datenbanken die Statistiken für alle Kurznamen, die dem lokalen Schema zugeordnet wurden.
 - Wählen Sie eine vorhandene Protokolldatei aus, oder geben Sie den vollständig qualifizierten Pfad für eine neue Protokolldatei ein.
 - b. Geben Sie auf der Seite **Zeitplan** an, wann die Aktualisierung für die Kurznamenstatistiken stattfinden soll.
 - Geben Sie zur Aktualisierung der Statistiken für einen einzelnen Kurznamen den gewünschten Aktualisierungszeitpunkt an.

Rufen Sie die gespeicherte Prozedur SYSPROC.NNSTAT auf, um diese Task von der Befehlszeile aus oder in der Anwendung auszuführen.

Zugehörige Konzepte:

- „Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht“ auf Seite 219

Zugehörige Tasks:

- „Anzeigen des Aktualisierungsstatus der Kurznamenstatistiken“ auf Seite 224

Zugehörige Referenzen:

- „Abrufen von Kurznamenstatistiken von der Befehlszeile aus - Beispiele“ auf Seite 222
- Kapitel 32, „Gespeicherte Prozedur SYSPROC.NNSTAT“, auf Seite 341

Abrufen von Kurznamenstatistiken von der Befehlszeile aus - Beispiele

Im folgenden Beispiel ruft der Server mit zusammengeschlossenen Datenbanken die Statistiken für alle Kurznamen auf dem Server DB2SERV ab; er erstellt dabei kein Protokoll.

```
CALL SYSPROC.NNSTAT('DB2SERV','NULL','NULL','NULL',?,?)
```

Im folgenden Beispiel ruft der Server mit zusammengeschlossenen Datenbanken die Statistiken für den Kurznamen STAFF im Schema ADMIN ab. Der Server mit zusammengeschlossenen Datenbanken schreibt das Protokoll in die Datei /home/iiuser/reportlogs/log1.txt.

```
CALL SYSPROC.NNSTAT(
  'NULL', 'ADMIN', 'STAFF', '/home/iiuser/reportlogs/log1.txt', ?, ?)
```

Im folgenden Beispiel ruft der Server mit zusammengeschlossenen Datenbanken Statistiken für alle Kurznamen auf dem Server DB2Serv im Schema admin ab. Der Server mit zusammengeschlossenen Datenbanken schreibt das Protokoll in die Datei /home/iiuser/stats/recent.log.

```
CALL SYSPROC.NNSTAT(
  'DB2Serv', 'admin', 'NULL', '/home/iiuser/stats/recent.log', ?, ?)
```

| **Zugehörige Konzepte:**

- | • „Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht“ auf Seite
| 219

| **Zugehörige Tasks:**

- | • „Abrufen von Kurznamenstatistiken“ auf Seite 220

| **Erstellen eines DB2-Toolskatalogs**

| Wenn Sie die Statistiken für einen Kurznamen aktualisieren, können Sie zur zeitli-
| chen Planung der Aktualisierung einen DB2-Toolskatalog verwenden. Die zeitliche
| Planung der Aktualisierung ist nur über die DB2-Steuerzentrale möglich. Wenn Sie
| über keinen DB2-Toolskatalog verfügen, werden Sie zur Erstellung eines Katalogs
| aufgefordert.

| **Voraussetzungen:**

| Der DB2-Verwaltungsserver muss installiert sein.

| **Vorgehensweise:**

| Ein DB2-Toolskatalog wird von der DB2-Steuerzentrale oder der Eingabeauf-
| forderung der Befehlszeile aus erstellt.

| Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszufüh-
| ren:

- | 1. Das Fenster **Statistik aktualisieren** wird angezeigt, wenn Sie Kurznamen-
| statistiken aktualisieren.
| 2. Wählen Sie das System aus, auf dem Sie eine Datenbank für den DB2-Toolska-
| talog erstellen möchten.

| Die Datenbank muss sich auf einem katalogisierten System befinden, das der-
| zeit über keinen Metadatenpeicher verfügt. Wenn das gewünschte System
| nicht katalogisiert ist, müssen Sie es vor der Erstellung der Datenbank für den
| DB2-Toolskatalog katalogisieren.

| **Zugehörige Tasks:**

- | • „Abrufen von Kurznamenstatistiken“ auf Seite 220
| • „Einrichten und Konfigurieren der Toolskatalogdatenbank und des DAS-Schedu-
| ler“ in *Systemverwaltung: Implementierung*

| **Zugehörige Referenzen:**

- | • „CREATE TOOLS CATALOG Command“ in *Command Reference*

Anzeigen des Aktualisierungsstatus der Kurznamenstatistiken

Nachdem Sie eine Aktualisierung der Statistiken für einen Kurznamen angefordert haben, können Sie den Status der Aktualisierung anzeigen. Sie können diese Informationen zur Auswertung der Abfrageleistung verwenden.

Vorgehensweise:

Sie können den Status der Aktualisierungen von Kurznamenstatistiken von der DB2-Steuerzentrale oder der DB2-Befehlszeile aus anzeigen.

Gehen Sie wie folgt vor, um diese Task von der DB2-Steuerzentrale aus auszuführen:

1. Wählen Sie die Kurznamen aus, deren Aktualisierungsstatus Sie anzeigen möchten:
 - Führen Sie folgende Schritte aus, um den Aktualisierungsstatus aller Kurznamen anzuzeigen, die einer bestimmten Serverdefinition zugeordnet wurden:
 - a. Erweitern Sie den Ordner **Objekte zusammengesetzter Datenbanken**.
 - b. Erweitern Sie den Ordner **Serverdefinitionen** mit den zu aktualisierenden Kurznamen.
 - c. Klicken Sie mit der rechten Maustaste auf den Namen der Serverdefinition, und wählen Sie **Statistik** aus.
 - Wenn Sie den Aktualisierungsstatus eines bestimmten Kurznamens anzeigen möchten, erweitern Sie den Ordner **Kurznamen**.
 - a. Wählen Sie den Kurznamen aus, den Sie aktualisieren möchten.
 - b. Klicken Sie mit der rechten Maustaste auf den Kurznamen, und wählen Sie **Statistik** aus.
2. Wählen Sie **Ergebnisse anzeigen** aus. Das Fenster **Statistiken - Ergebnisse anzeigen** wird geöffnet.

Konsultieren Sie die Tabelle SYSPROC.FED_STATS, in der Sie ein Aktualisierungsprotokoll finden, um diese Task von der Befehlszeile aus auszuführen.

Zugehörige Konzepte:

- „Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht“ auf Seite 219

Zugehörige Tasks:

- „Abrufen von Kurznamenstatistiken“ auf Seite 220

Zugehörige Referenzen:

- „Abrufen von Kurznamenstatistiken von der Befehlszeile aus - Beispiele“ auf Seite 222

Teil 4. Anwendungsprogrammierung

Kapitel 17. Anwendungsprogrammierungsszenario

Informationen dazu, wie Sie DB2 Information Integrator zusammen mit IBM Websphere Portal verwenden können, um Anwendungen zu entwickeln, finden Sie in den folgenden Referenzen:

- *IBM Websphere Portal and DB2 Information Integrator*, unter der Webadresse <http://www.redbooks.ibm.com/redpieces/pdfs/sg246433.pdf>
- Beispielcode für das Szenario unter der Webadresse <ftp://www.redbooks.ibm.com/redbooks/SG246433/>

Kapitel 18. Anwendungsprogrammierung für Systeme zusammengesetzter Datenbanken

In diesem Kapitel werden die Punkte erörtert, die Programmierer bei der Entwicklung von Anwendungen für Systeme zusammengesetzter Datenbanken beachten sollten.

Detaillierte Informationen zur Anwendungsprogrammierung finden Sie in folgenden Handbüchern:

- *IBM DB2 Universal Database Application Development Guide: Building and Running Applications Version 8*
- *IBM DB2 Universal Database Application Development Guide: Programming Client Applications Version 8*

Interaktion von Clientanwendungen mit Datenquellen

Für Clientanwendungen erscheinen die Datenquellen in einem System zusammengesetzter Datenbanken als eine einzige kollektive Datenbank. Um Daten von Datenquellen abzurufen, übergeben Anwendungen Anforderungen an die zusammengesetzte Datenbank in DB2[®]-SQL. Anschließend verteilt DB2 UDB die Abfragen an die entsprechenden Datenquellen und gibt die Daten entweder an die Anwendungen zurück oder führt die angeforderte Aktion durch. Die zusammengesetzte Datenbank kann Daten aus lokalen Tabellen und Daten aus fernen Datenquellen so miteinander verknüpfen, als würde es sich nur um lokale Daten handeln. Sie können beispielsweise Daten einer lokalen Tabelle von DB2 für Windows[®], einer Informix[®]-Tabelle und einer Sybase-Sicht in einer einzigen SQL-Anweisung verknüpfen. Indem das System zusammengesetzter Datenbanken SQL-Anweisungen so verarbeitet, als wären die Datenquellen normale relationale Tabellen oder Sichten innerhalb der zusammengesetzten Datenbank, kann das System relationale Daten und nicht relationale Daten miteinander verknüpfen.

In einem System zusammengesetzter Datenbanken erfolgt der Zugriff auf Datenquellen durch Kurznamen. Ein *Kurzname* ist eine Kennung, die eine Anwendung verwendet, um auf ein Datenquellenobjekt wie z. B. eine Tabelle oder eine Sicht zu verweisen. Zum Schreiben an eine Datenquelle - z. B. zum Aktualisieren einer Datenquellentabelle - kann eine Anwendung DB2-SQL (mit Kurznamen) verwenden. Alternativ können Anwendungen in einer speziellen Sitzung, genannt *Durchgriff*, die SQL-Programmversion der Datenquelle (ohne Kurznamen) verwenden, um direkt auf die Datenquelle zuzugreifen.

Anwendungen, die DB2-SQL und Kurznamen verwenden, können auf jeden beliebigen Datentyp zugreifen, der von DB2 UDB erkannt wird.

Der Systemkatalog der zusammengesetzten Datenbank enthält Informationen über Objekte in der zusammengesetzten Datenbank und über Objekte in den Datenquellen. Da der Katalog Informationen über das gesamte System zusammengesetzter Datenbanken enthält, wird er als *globaler Katalog* bezeichnet.

Zugehörige Konzepte:

- „Direktes Abfragen von Datenquellen im Durchgriffsmodus“ auf Seite 242

Zugehörige Tasks:

- „Verweisen auf Datenquellenobjekte in SQL-Anweisungen über Kurznamen“ auf Seite 230
- „Kurznamen in DDL-Anweisungen“ auf Seite 231
- „Datenquellenstatistiken - Anwendungsauswirkungen“ auf Seite 232
- „Kurznamen zum Aufrufen gespeicherter Prozeduren“ auf Seite 233
- „Definieren von Spaltenoptionen für Kurznamen“ auf Seite 233
- „Zugreifen auf Datenquellen mit Durchgriffssitzungen“ auf Seite 123

Arbeiten mit Kurznamen in Ihren Anwendungen

Verweisen auf Datenquellenobjekte in SQL-Anweisungen über Kurznamen

Bei einem System zusammenschlossener Datenbanken müssen Sie in Ihren SQL-Anweisungen den Datenquellenserver, das Schema und das Objekt nicht angeben. Verwenden Sie stattdessen die Kurznamen, die für Datenquellenobjekte in Ihren SQL-Anweisungen definiert wurden, um Datenquellenobjekte abzufragen.

Für Datenquellenobjekte müssen in der zusammenschlossenen Datenbank Kurznamen registriert werden, bevor sie in Abfragen aufgenommen werden können.

Kurznamen in den Anweisungen SELECT, INSERT, UPDATE und DELETE verwenden:

Angenommen, Sie definieren den Kurznamen NFXDEPT, der eine Informix-Tabelle mit dem Namen PERSON.DEPT darstellt, wobei Folgendes gilt:

- PERSON ist das Datenquellenschema.
- DEPT ist der Name der Datenquellentabelle.

Die Anweisung `SELECT * FROM NFXDEPT` ist für den Server mit zusammenschlossenen Datenbanken zulässig. Die Anweisung `SELECT * FROM PERSON.DEPT` ist jedoch nicht zulässig (außer bei einer Durchgriffssitzung). Für den Server mit zusammenschlossenen Datenbanken ist `PERSON.DEPT` nicht als Kurzname registriert.

Kurznamen in der Anweisung CREATE TABLE verwenden:

Angenommen, Sie wollen eine lokale gespeicherte Abfragetabelle (MQT, Materialized Query Table) auf der Basis einer Tabelle erstellen, für die Sie einen Kurznamen definiert haben. Die Anweisung `CREATE TABLE` lautet beispielsweise:

```
CREATE TABLE table_name LIKE nickname
```

Zugehörige Konzepte:

- „Direktes Abfragen von Datenquellen im Durchgriffsmodus“ auf Seite 242

Zugehörige Tasks:

- „Überlegungen und Einschränkungen bei der Verwendung des Durchgriffsmodus in Systemen zusammenschlossener Datenbanken“ auf Seite 243
- „Durchgriffssitzungen zu Oracle-Datenquellen“ auf Seite 244

Kurznamen in DDL-Anweisungen

Für Datenquellenobjekte müssen in der zusammengeschlossenen Datenbank Kurznamen registriert werden, bevor sie in DDL-Anweisungen aufgenommen werden können. Im Folgenden sind verschiedene Beispiele für DDL-Anweisungen aufgeführt, die in Systemen zusammengeschlossener Datenbanken verwendet werden können:

Kurznamen in der Anweisung COMMENT ON verwenden:

Mit der Anweisung COMMENT ON werden im globalen Katalog der zusammengeschlossenen Datenbanken Kommentare hinzugefügt oder ersetzt. Die Anweisung COMMENT ON ist für einen Kurznamen und Spalten gültig, die für einen Kurznamen definiert wurden. Mit dieser Anweisung werden keine Datenquellenkataloge aktualisiert.

Kurznamen in den Anweisungen GRANT und REVOKE verwenden:

Die Anweisungen GRANT und REVOKE sind mit einem Kurznamen für bestimmte Zugriffsrechte sowie für alle Benutzer und Gruppen zulässig. DB2 UDB gibt jedoch keine entsprechende Anweisung GRANT oder REVOKE für das Objekt in der Datenquelle aus, auf die der Kurzname verweist.

Angenommen, der Benutzer JON erstellt für eine Oracle-Tabelle, die keinen Index hat, einen Kurznamen. Der Kurzname ist ORAREM1. Später definiert der Oracle-Datenbankadministrator für diese Tabelle einen Index. Die Benutzerin EILEEN möchte jetzt, dass die zusammengeschlossene DB2-Datenbank weiß, dass dieser Index vorhanden ist, damit das Abfrageoptimierungsprogramm Strategien für einen effizienteren Zugriff auf die Tabelle entwickeln kann. EILEEN kann die zusammengeschlossene Datenbank darüber informieren, dass ein neuer Index vorhanden ist, indem sie für ORAREM1 eine Indexspezifikation erstellt.

Die Informationen zum Index werden in der Katalogsicht SYSSTAT.INDEXES gespeichert. Verwenden Sie die Anweisung GRANT, um EILEEN das Indexzugriffsrecht für diesen Kurznamen zu erteilen, damit sie die Indexspezifikation erstellen kann.

```
GRANT INDEX ON NICKNAME ORAREM1 TO USER EILEEN
```

Um die Zugriffsrechte der Benutzerin EILEEN zur Erstellung einer Indexspezifikation für den Kurznamen ORAREM1 zu widerrufen, muss die Anweisung REVOKE verwendet werden:

```
REVOKE INDEX ON ORAREM1 FROM USER EILEEN
```

Zugehörige Tasks:

- „Erstellen von Indexspezifikationen für Datenquellenobjekte“ auf Seite 84
- „Mit Kurznamen verwendbare SQL-Anweisungen“ auf Seite 116

Zugehörige Referenzen:

- „COMMENT statement“ in *SQL Reference, Volume 2*
- „GRANT (Table, View, or Nickname Privileges) statement“ in *SQL Reference, Volume 2*
- „REVOKE (Table, View, or Nickname Privileges) statement“ in *SQL Reference, Volume 2*

Datenquellenstatistiken - Anwendungsauswirkungen

Wenn für ein Datenquellenobjekt ein Kurzname erstellt wird, wird der globale Katalog der zusammengeschlossenen Datenbank mit Informationen zu diesem Objekt aktualisiert. Das DB2-Abfrageoptimierungsprogramm verwendet diese Informationen dazu, das Abrufen von Daten aus dem betreffenden Objekt zu planen. Es ist wichtig, sich zu vergewissern, dass die Datenquelleninformationen aktuell sind. Die zusammengeschlossene Datenbank findet Änderungen an Datenquellenobjekten nicht automatisch.

Im globalen Katalog gespeicherte Katalogstatistiken:

Welche Informationen im globalen Katalog zu einem Datenquellenobjekt gespeichert werden, hängt vom Typ des Objekts ab. Bei Datenbanktabellen und -sichten werden der Name des Objekts, die Spaltennamen und -attribute im globalen Katalog gespeichert.

Bei Tabellen werden außerdem folgende Informationen gespeichert:

- Statistiken. Beispielsweise sind dies die Anzahl der Zeilen und die Anzahl der Seiten, auf denen die Zeilen bestehen. Stellen Sie sicher, dass DB2 UDB die aktuellen Statistiken abrufen. Führen Sie den Datenquellenbefehl für die Tabelle aus, der dem Befehl RUNSTATS entspricht, bevor Sie den Kurznamen erstellen.
- Indexbeschreibungen. Wenn die Tabelle über keine Indizes verfügt, können Sie für den Katalog Metadaten bereitstellen, die normalerweise in einer Indexdefinition enthalten sind. Beispielsweise können Sie den Katalog informieren, welche Spalte(n) in der Tabelle eindeutige Werte enthält/enthalten und ob Zeilen eindeutig sind. Diese Metadaten, die in ihrer Gesamtheit als *Indexspezifikation* bezeichnet werden, können Sie generieren, indem Sie die Anweisung CREATE INDEX absetzen und auf den Kurznamen der Tabelle verweisen. Sie können mit der Anweisung CREATE INDEX die Klausel SPECIFICATION ONLY verwenden, wenn Sie nur eine Indexspezifikation und keinen tatsächlichen Index erstellen wollen.

Um zu ermitteln, welche Datenquelleninformationen im globalen Katalog gespeichert sind, müssen Sie die Katalogsichten SYSCAT.TABLES und SYSCAT.COLUMNS abfragen. Um zu ermitteln, welche Indexinformationen für Datenquellen im Katalog gespeichert sind oder welcher Inhalt in einer bestimmten Indexspezifikation enthalten ist, müssen Sie die Katalogsicht SYSCAT.INDEXES abfragen.

Anwendungen ändern, sodass sie statt auf die Sicht SYSCAT auf die Sicht SYSSTAT verweisen:

Die SYSCAT-Sichten der DB2 Version 8 sind nun schreibgeschützt. Wenn Sie eine UPDATE- oder INSERT-Operation für eine Sicht im Schema SYSCAT ausführen, schlägt diese fehl. Es wird empfohlen, den Systemkatalog mit Hilfe der SYSSTAT-Sichten zu aktualisieren. Ändern Sie auf die Sicht SYSCAT verweisende Anwendungen, sodass diese stattdessen auf die Sicht SYSSTAT verweisen.

Zugehörige Konzepte:

- „Katalogstatistiken“ in *Systemverwaltung: Optimierung*
- „Katalogstatistiktabellen“ in *Systemverwaltung: Optimierung*
- „Die zusammengeschlossene Datenbank“ auf Seite 8

Zugehörige Tasks:

- „Erstellen von Indexspezifikationen für Datenquellenobjekte“ auf Seite 84
- „Kurznamenkenndaten mit Auswirkungen auf die globale Optimierung“ auf Seite 173

Zugehörige Referenzen:

- „SYSCAT.COLUMNS catalog view“ in *SQL Reference, Volume 1*
- „SYSCAT.INDEXES catalog view“ in *SQL Reference, Volume 1*
- „SYSCAT.TABLES catalog view“ in *SQL Reference, Volume 1*
- Kapitel 19, „Sichten in der globalen Katalogtabelle mit Informationen der zusammengesetzten Datenbank“, auf Seite 247

Kurznamen zum Aufrufen gespeicherter Prozeduren

Wenn Sie Anwendungen von DataJoiner migrieren, die gespeicherte Prozeduren über Kurznamen aufrufen, müssen Sie Ihre Anwendungen möglicherweise ändern. DB2 Information Integrator bietet momentan keine Unterstützung für das Aufrufen von gespeicherten Prozeduren unter Verwendung von Kurznamen.

Definieren von Spaltenoptionen für Kurznamen

Spaltenoptionen sind Parameter in den Anweisungen CREATE NICKNAME und ALTER NICKNAME. Sie können Spaltenoptionen angeben, wenn Sie mit der Erstellung eines Kurznamens beginnen oder wenn Sie einen bestehenden Kurznamen ändern.

Die Informationen, die Sie über die Spaltenoptionen bereitstellen, werden im globalen Katalog gespeichert.

Nicht relationale Datenquellen

Spaltenoptionen sind für jeden nicht relationalen Wrapper eindeutig. Diese Optionen werden in der Regel beim Absetzen der Anweisung CREATE NICKNAME gesetzt.

Relationale Datenquellen

Für relationale Datenquellen können zwei Spaltenoptionen verwendet werden: NUMERIC_STRING und VARCHAR_NO_TRAILING_BLANKS.

Definieren der Spaltenoption NUMERIC_STRING

Wenn in einer Datenquelle eine Spalte mit numerischen Zeichenfolgen nur numerische und keine weiteren Zeichen (einschließlich Leerzeichen) enthält, müssen Sie die Spaltenoption NUMERIC_STRING auf 'Y' setzen. Dadurch können Abfragen, die diese Spalte verwenden, bei Sortier- und Vergleichsoperationen optimiert werden. Beispiel:

```
ALTER NICKNAME nickname
ALTER COLUMN local_column_name
OPTIONS (SET NUMERIC_STRING 'Y')
```

Angeben der Spaltenoption VARCHAR_NO_TRAILING_BLANKS

Einige Datenquellen, wie z. B. Oracle, verfügen nicht über eine Zeichenfolgevergleichslogik mit Leerzeichen als Füllzeichen wie sie in DB2 für Linux, UNIX und Windows verwendet wird. Davon sind Datentypen wie VARCHAR und VARCHAR2 betroffen. Daher müssen Vergleichselemente, die diese Datentypen verwenden, vom Abfrageoptimierungsprogramm erneut geschrieben werden, um konsistente Abfrageergebnisse zu gewährleisten. Das erneute Schreiben der

Anweisungen kann sich auf die Leistung auswirken. Wenn Sie diese Option für eine bestimmte Spalte angeben, erhält das Abfrageoptimierungsprogramm dadurch Informationen zu diesen Spalten, die ihm das Generieren effizienterer SQL-Anweisungen ermöglichen.

Beispiel:

```
ALTER NICKNAME nickname
  ALTER COLUMN local_column_name
  OPTIONS (SET VARCHAR_NO_TRAILING_BLANKS 'Y')
```

Zugehörige Referenzen:

- Kapitel 24, „Kurznamenspaltenoptionen für Systeme zusammengesetzter Datenbanken“, auf Seite 283

Erstellen und Verwenden von Sichten für zusammengesetzte Datenbanken

Eine Sicht in der zusammengesetzten Datenbank, deren Basistabellen sich in fernen Datenquellen befinden, wird als *Sicht für zusammengesetzte Datenbanken* bezeichnet. Auf diese Basistabellen wird in der Sicht für zusammengesetzte Datenbanken durch Kurznamen und nicht durch die Tabellennamen der Datenquelle verwiesen.

Die Vorteile der Verwendung von Sichten für zusammengesetzte Datenbanken ähneln denen der Verwendung von Sichten, die für mehrere lokale Tabellen in einem zentralen Manager für relationale Datenbanken definiert wurden:

- Sichten bieten eine integrierte Darstellung der Daten.
- Sie können Tabellenspalten, die vertrauliche oder sensible Daten enthalten, aus einer Sicht ausschließen.

Einschränkungen:

Auf Sichten für zusammengesetzte Datenbanken, die von mehreren Datenquellenobjekten erstellt wurden, besteht nur Lesezugriff. Sie können auch nicht aktualisiert werden.

Auf Sichten für zusammengesetzte Datenbanken, die aus nur einem Datenquellenobjekt erstellt wurden, kann (muss jedoch nicht) nur Lesezugriff bestehen.

- Auf eine Sicht für zusammengesetzte Datenbanken, die von einer einzelnen nicht relationalen Datenquelle erstellt wurde, besteht nur Lesezugriff.
- Eine Sicht für zusammengesetzte Datenbanken, die aus einer einzelnen relationalen Datenquelle erstellt wurde, kann Aktualisierungen zulassen. Dies ist abhängig vom Inhalt der Anweisung CREATE VIEW.

Vorgehensweise:

Sie erstellen eine Sicht für zusammengesetzte Datenbanken für Datenquellenobjekte, die über Kurznamen verfügen. Die Aktion, eine Sicht für zusammengesetzte Datenbanken für Datenquellendaten zu erstellen, wird in manchen Fällen als „Erstellen einer Sicht für einen Kurznamen“ bezeichnet. Dieser Ausdruck gibt die Tatsache wieder, dass für die Erstellung der Sicht für zusammengesetzte Datenbanken die Gesamtauswahl der Anweisung CREATE VIEW auf die Kurznamen aller Datenquellentabellen und -sichten verweisen muss, die in der Sicht für zusammengesetzte Datenbanken enthalten sein sollen.

Beispiel: Sicht für zusammengeschlossene Datenbanken für Mischung ähnlicher Daten aus verschiedenen Datenquellenobjekten erstellen:

Angenommen, Sie verfügen über Kundendaten auf drei verschiedenen Servern. Ein Server befindet sich in Europa, einer in Asien und einer in Südamerika. Die Kundendaten für Europa befinden sich in einer Oracle-Tabelle. Der Kurzname für diese Tabelle lautet `ORA_EU_CUST`. Die Kundendaten für Asien befinden sich in einer Sybase-Tabelle. Der Kurzname für diese Tabelle lautet `SYB_AS_CUST`. Die Kundendaten für Südamerika befinden sich in einer Informix-Tabelle. Der Kurzname für diese Tabelle lautet `INFMX_SA_CUST`. Jede Tabelle hat Spalten für die Kundennummer (`CUST_NO`), den Kundennamen (`CUST_NAME`), die Produktnummer (`PROD_NO`) und die Bestellmenge (`QUANTITY`). Die Syntax zum Erstellen einer Sicht von diesen drei Kurznamen, in der diese Kundendaten zusammengefügt werden, lautet wie folgt:

```
CREATE VIEW FV1
  AS SELECT * FROM ORA_EU_CUST
UNION
  SELECT * FROM SYB_AS_CUST
UNION
  SELECT * FROM INFMX_SA_CUST
```

Beispiel: Daten zum Erstellen einer Sicht für zusammengeschlossene Datenbanken verknüpfen:

Angenommen, Sie haben auf einem Server Kundendaten und auf einem anderen Server Verkaufsdaten gespeichert. Die Kundendaten befinden sich in einer Oracle-Tabelle. Der Kurzname für diese Tabelle lautet `ORA_EU_CUST`. Die Verkaufsdaten befinden sich in einer Sybase-Tabelle. Der Kurzname für diese Tabelle lautet `SYB_SALES`. Sie möchten die Kundendaten mit den Käufen dieser Kunden abgleichen. Jede Tabelle hat eine Spalte für die Kundennummer (`CUST_NO`). Die Syntax zum Erstellen einer Sicht für zusammengeschlossene Datenbanken von diesen zwei Kurznamen, die diese Daten miteinander verknüpft, lautet wie folgt:

```
CREATE VIEW FV4
  AS SELECT A.CUST_NO, A.CUST_NAME, B.PROD_NO, B.QUANTITY
  FROM ORA_EU_CUST A, SYB_SALES B
  WHERE A.CUST_NO=B.CUST_NO
```

Zugehörige Tasks:

- „Zugreifen auf heterogene Daten durch Sichten für zusammengeschlossene Datenbanken“ auf Seite 124

Zugehörige Referenzen:

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

Verwenden von Isolationsstufen zur Gewährleistung der Datenintegrität

Eine Isolationsstufe, die einem Anwendungsprozess zugeordnet ist, definiert den Grad der Isolation dieses Anwendungsprozesses gegenüber anderen Anwendungsprozessen, die gleichzeitig ausgeführt werden. Die Isolationsstufe wird als Attribut eines Pakets angegeben, das für die Anwendungsprozesse gilt, die dieses Paket verwenden. Isolationsstufen werden verwendet, wenn eine Anwendung vorbereitet oder gebunden wird.

Sperren tritt für die Basistabellenzeile auf. Der Datenbankmanager kann jedoch mehrere Zeilensperren durch eine einzige Tabellensperre ersetzen. Dies wird als Sperreneskulation bezeichnet. Für einen Anwendungsprozess ist zumindest die minimale angeforderte Sperrstufe garantiert.

Sie können die Datenintegrität für eine Datenquellentabelle beibehalten, indem Sie anfordern, dass die Tabellenzeilen bei einer bestimmten Isolationsstufe gesperrt werden. Um beispielsweise sicherzustellen, dass Sie als einziger Benutzer Zugriff auf eine Zeile haben, müssen Sie für diese die Isolationsstufe 'Wiederholtes Lesen' (Repeatable Read, RR) angeben.

Der Server mit zusammengeschlossenen Datenbanken ordnet die angeforderte Isolationsstufe einer entsprechenden Isolationsstufe in der Datenquelle zu. Bei jeder Verbindung zur Datenquelle bestimmt der Wrapper die Isolationsstufe von DB2 für Linux, UNIX und Windows. Die Isolationsstufe der fernen Datenquelle wird auf die entsprechende Stufe gesetzt. Wenn keine exakte Entsprechung vorhanden ist, legt der Server mit zusammengeschlossenen Datenbanken die Isolationsstufe auf die nächsthöhere Stufe fest. Sobald eine Verbindung zur Datenquelle hergestellt wurde, kann die Isolationsstufe für die Dauer der Verbindung nicht mehr geändert werden.

Folgende Isolationsstufen können angegeben werden:

- CS Cursorstabilität (Cursor Stability)
- RR Wiederholtes Lesen (Repeatable Read)
- RS Lesestabilität (Read Stability)
- UR Nicht festgeschriebener Lesevorgang (Uncommitted Read)

Vorgehensweise:

In der folgenden Tabelle sind die Isolationsstufen aufgelistet, die Sie für die unterstützten Datenquellen anfordern können.

Tabelle 17. Vergleichbare Isolationsstufen für den Server mit zusammengeschlossenen Datenbanken und unterstützte Datenquellen.

DB2-Server mit zusammengeschlossenen Datenbanken	Cursorstabilität (Cursor Stability)	Wiederholtes Lesen (Repeatable Read)	Lesestabilität (Read Stability)	Nicht festgeschriebener Lesevorgang (Uncommitted Read)
DB2-Produktfamilie	Cursorstabilität (Cursor Stability)	Wiederholtes Lesen (Repeatable Read)	Lesestabilität (Read Stability)	Nicht festgeschriebener Lesevorgang (Uncommitted Read)
Informix	Cursorstabilität (Cursor Stability)	Wiederholtes Lesen (Repeatable Read)	Wiederholtes Lesen (Repeatable Read)	Fehlerhafte Leseoperation
Microsoft SQL Server	Read Committed	Serializable	Wiederholtes Lesen (Repeatable Read)	Read Uncommitted

Tabelle 17. Vergleichbare Isolationsstufen für den Server mit zusammengeschlossenen Datenbanken und unterstützte Datenquellen. (Forts.)

DB2-Server mit zusammengeschlossenen Datenbanken	Cursorstabilität (Cursor Stability)	Wiederholtes Lesen (Repeatable Read)	Lesestabilität (Read Stability)	Nicht festgeschriebener Lesevorgang (Uncommitted Read)
ODBC	Read Committed	Serializable	Wiederholtes Lesen (Repeatable Read)	Read Uncommitted
Oracle	Read Committed	Serializable	Serializable	Read Committed
Sybase	Stufe 1	Stufe 3	Stufe 3	Stufe 0

OLE DB-, Teradata- und nicht relationale Datenquellen verfügen über kein vergleichbares Konzept für Isolationsstufen, wie es bei DB2 für Linux, UNIX und Windows verwendet wird. Zwischen den DB2-Isolationsstufen und den OLE DB-, Teradata- und den nicht relationalen Datenquellen gibt es keine Zuordnungen.

Zugehörige Konzepte:

- „Isolation levels“ in *SQL Reference, Volume 1*
- „Applications in Host or iSeries Environments“ in *Application Development Guide: Programming Client Applications*
- „Interaktion von Clientanwendungen mit Datenquellen“ auf Seite 229

LOB-Unterstützung in einem System zusammengeschlossener Datenbanken

Ein System zusammengeschlossener Datenbanken ermöglicht es Ihnen, auf große Objekte (Large Objects = LOBs) in fernen Datenquellen zuzugreifen und diese zu ändern. Da LOBs sehr groß sein können, ist ihre Übertragung von einer fernen Datenquelle unter Umständen sehr zeitaufwändig. Die zusammengeschlossene DB2®-Datenbank versucht, die Übertragung von LOB-Daten von Datenquellen zu minimieren und angeforderte LOB-Daten ohne Speichern in DB2 UDB direkt von der Datenquelle an die anfordernde Anwendung zu übergeben.

Ein System zusammengeschlossener Datenbanken unterstützt die Ausführung von SELECT-Operationen für LOBs in DRDA®, Informix-, Microsoft® SQL Server-, Oracle- und Sybase-Datenquellen. Beispiel:

```
SELECT empname, picture FROM infmx_emp_table
WHERE empno = '01192345'
```

Hierbei steht *picture* für eine LOB-Spalte und *infmx_emp_table* für einen Kurznamen, der auf eine Informix™-Tabelle mit Mitarbeiterdaten verweist.

Ein System zusammengeschlossener Datenbanken unterstützt SELECT-, INSERT-, UPDATE- und DELETE-Operationen für LOBs in Oracle-Datenquellen (ab Version 8).

Die von DB2 für Linux, UNIX® und Windows® Version 8 unterstützten Lese- und Schreiboperationen sind in der folgenden Tabelle aufgeführt:

Tabelle 18. Lese- und Schreibunterstützung für LOBs

Datenquelle	Typ der Operation
DB2-Produktfamilie ¹	Lesezugriff
BioRS	Lesezugriff
BLAST	Lesezugriff und internes Binden
Entrez	Lesezugriff
HMMER	Lesezugriff und internes Binden
Informix	Lesezugriff
Microsoft SQL Server	Lesezugriff
Oracle (NET8-Wrapper)	Lese- und Schreibzugriff
ODBC	Lesezugriff
Sybase	Lesezugriff
Teradata	Lesezugriff
Web-Services	Lesezugriff und externes Binden nur für CLOB
WebSphere [®] Business Integration	Lesezugriff und externes Binden nur für CLOB
XML	Lesezugriff

Anmerkungen:

1. Für die LOB-Unterstützung ist DB2 UDB für iSeries™ ab Version 5 erforderlich. DB2 Information Integrator Version 8 kann nicht auf LOB-Daten von DB2 UDB für Linux, UNIX und Windows Version 7 zugreifen.

Teradata-LOBs

Zwischen Teradata-LOBs und DB2-LOBs bestehen leichte Unterschiede. Teradata verfügt über keine Datentypen von der Größe der von DB2 UDB unterstützten LOBs. Einige Teradata-Datentypen können jedoch bis zu 64000 Byte lang sein. Dabei handelt es sich um die Datentypen CHAR, VARCHAR, BYTE, VARBYTE, GRAPHIC und VARGRAPHIC. Diese Teradata-Datentypen werden DB2 LOB-Datentypen zugeordnet, wenn die Länge des Teradata-Datentyps die Grenzwerte des entsprechenden DB2-Datentyps überschreitet.

LOB-Längen

Einige Datenquellen, z. B. Oracle und Informix, speichern die Längen von LOB-Spalten nicht in ihren Systemkatalogen. Wenn Sie einen Kurznamen für eine Tabelle erstellen, werden Informationen aus dem Systemkatalog abgerufen, einschließlich der Spaltenlänge. Da für die LOB-Spalten keine Längenangabe vorhanden ist, nimmt die zusammengeschlossene Datenbank an, dass die Länge der maximalen Länge einer LOB-Spalte in DB2 für Linux, UNIX und Windows entspricht. Die zusammengeschlossene Datenbank speichert die in DB2 für Linux, UNIX und Windows maximal mögliche Länge als Länge der Kurznamenspalte im Katalog der zusammengeschlossenen Datenbank.

Zugehörige Konzepte:

- „LOB-Querverweise“ auf Seite 239
- „Einschränkungen für LOBs“ auf Seite 239

Zugehörige Referenzen:

- Kapitel 27, „Standardmäßige progressive Datentypzuordnungen“, auf Seite 299

LOB-Unterstützung in einem System zusammengeschlossener Datenbanken - ausführliche Informationen

LOB-Querverweise

Anwendungen können Querverweise auf in fernen Datenquellen gespeicherte LOBs anfordern. Ein *Querverweis* ist ein in einer Hostvariablen gespeicherter Wert von 4 Byte. Eine Anwendung kann unter Verwendung eines Querverweises auf einen LOB-Wert (oder einen LOB-Ausdruck) im Datenbanksystem verweisen. Eine Anwendung kann unter Verwendung eines Querverweises den LOB-Wert so verändern, als wäre der LOB-Wert in einer regulären Hostvariablen gespeichert. Bei der Verwendung von Querverweisen muss der LOB-Wert nicht vom Server der Datenquelle und möglicherweise wieder zurück transportiert werden.

DB2[®] UDB kann LOBs von fernen Datenquellen abrufen, auf dem Server mit zusammengeschlossenen Datenbanken speichern und dann einen LOB-Querverweis für das gespeicherte LOB ausgeben. LOB-Querverweise werden in folgenden Fällen ausgegeben:

- Eine Anwendung gibt eine SQL-Anweisung FREE LOCATOR aus.
- Eine Anwendung gibt eine Anweisung COMMIT aus.
- Das DB2-Exemplar für zusammengeschlossene Datenbanken wird erneut gestartet.

Zugehörige Konzepte:

- „Large object locators“ in *Application Development Guide: Programming Server Applications*
- „LOB-Unterstützung in einem System zusammengeschlossener Datenbanken“ auf Seite 237
- „Einschränkungen für LOBs“ auf Seite 239

Einschränkungen für LOBs

In Systemen zusammengeschlossener Datenbanken gelten für LOBs die folgenden Einschränkungen:

- Die zusammengeschlossene Datenbank kann ferne LOBs nicht an eine Dateireferenzvariable binden.
- LOBs werden in Durchgriffssitzungen nicht unterstützt.

Zugehörige Konzepte:

- „LOB-Unterstützung in einem System zusammengeschlossener Datenbanken“ auf Seite 237
- „LOB-Querverweise“ auf Seite 239

Verteilte Anforderungen zur Abfrage von Datenquellen

Abfragen, die an die zusammengeschlossene Datenbank übergeben werden, können Ergebnisse von einer einzigen Datenquelle anfordern. Normalerweise werden jedoch Ergebnisse von mehreren Datenquellen angefordert. Da eine typische Abfrage an mehrere Datenquellen verteilt wird, wird sie *verteilte Anforderung* genannt. Im Allgemeinen verwendet eine verteilte Anforderung mindestens eine

von drei SQL-Konventionen, um anzugeben, wo Daten abgerufen werden sollen (Unterabfragen, Gruppenoperatoren und Verknüpfungsunterauswahlelementen).

Angenommen, Sie haben einen Server mit zusammengeschlossenen Datenbanken, der für den Zugriff auf eine Datenquelle von DB2 für OS/390, eine Datenquelle von DB2 für iSeries und eine Datenquelle von Oracle konfiguriert ist. In jeder dieser Datenquellen ist eine Tabelle gespeichert, die Mitarbeiterinformationen enthält. Der Server mit zusammengeschlossenen Datenbanken verweist auf diese Tabellen über Kurznamen, die auf die Position zeigen, an der sie gespeichert sind.

UDB390_EMPLOYEES

Kurzname für eine Tabelle in einer Datenquelle von DB2 für OS/390, die Mitarbeiterinformationen enthält.

iSERIES_EMPLOYEES

Kurzname für eine Tabelle in einer Datenquelle von DB2 für iSeries, die Mitarbeiterinformationen enthält.

ORA_EMPLOYEES

Kurzname für eine Tabelle in einer Datenquelle von Oracle, die Mitarbeiterinformationen enthält.

ORA_REGIONS

Kurzname für eine Tabelle in einer Datenquelle von Oracle, die Informationen zu den Regionen enthält, in denen die Mitarbeiter leben.

Die folgenden Beispiele zeigen die drei SQL-Konventionen, die mit verteilten Anforderungen verwendet werden. Dabei werden die Kurznamen benutzt, die für alle Tabellen definiert wurden.

Beispiel: Verteilte Anforderung mit Unterabfrage:

Die Tabelle `iSERIES_EMPLOYEES` enthält die Telefonnummern von Mitarbeitern, die in Asien leben. Sie enthält außerdem die Regionscodes, die diesen Telefonnummern zugeordnet sind, listet jedoch nicht die Regionen auf, die die Codes darstellen. Die Tabelle `ORA_REGIONS` listet sowohl Codes als auch Regionen auf. Die folgende Abfrage verwendet eine Unterabfrage, um den Regionscode für China zu ermitteln. Danach wird dieser Regionscode dazu verwendet, eine Liste mit den Mitarbeitern in der Tabelle `iSERIES_EMPLOYEES` zurückzugeben, die eine Telefonnummer in China haben.

```
SELECT name, telephone FROM db2admin.iSERIES_employees
WHERE region_code IN
  (SELECT region_code FROM dbadmin.ora_regions
   WHERE region_name = 'CHINA')
```

Beispiel: Verteilte Anforderung mit Gruppenoperatoren:

Der Server mit zusammengeschlossenen Datenbanken unterstützt drei Gruppenoperatoren: UNION, EXCEPT und INTERSECT.

- Verwenden Sie den Gruppenoperator UNION, um die Zeilen zu kombinieren, die eine beliebige von mindestens zwei SELECT-Anweisungen erfüllen.
- Verwenden Sie den Gruppenoperator EXCEPT, um die Zeilen abzurufen, die die erste Anweisung SELECT erfüllen, nicht aber die zweite.
- Verwenden Sie den Gruppenoperator INTERSECT, um die Zeilen abzurufen, die beide SELECT-Anweisungen erfüllen.

Alle drei Gruppenoperatoren können den Operanden ALL verwenden, um anzugeben, dass doppelte Zeilen nicht aus dem Ergebnis entfernt werden sollen. Dadurch ist kein weiterer Sortiervorgang erforderlich.

Durch die folgende Abfrage werden alle Mitarbeiternamen und Regionscodes abgerufen, die sowohl in der Tabelle iSERIES_EMPLOYEES als auch in der Tabelle UDB390_EMPLOYEES enthalten sind, obwohl sich die Tabellen in unterschiedlichen Datenquellen befinden.

```
SELECT name, region_code
FROM as400_employees
INTERSECT
SELECT name, region_code
FROM udb390_employees
```

Beispiel: Verteilte Anforderung für eine Verknüpfung:

Eine relationale Verknüpfung generiert eine Ergebnismenge, die eine Kombination aus Spalten enthält, die aus mindestens zwei Tabellen abgerufen wurden. Sie sollten Bedingungen angeben, durch die die Größe der Zeilen in der Ergebnismenge eingeschränkt wird.

Die nachfolgende Abfrage kombiniert Mitarbeiternamen und ihre zugehörigen Regionsnamen, indem die in zwei Tabellen aufgelisteten Regionscodes verglichen werden. Jede Tabelle befindet sich in einer anderen Datenquelle.

```
SELECT t1.name, t2.region_name
FROM dbadmin.iSERIES_employees t1, dbadmin.ora_regions t2
WHERE t1.region_code = t2.region_code
```

Optimieren verteilter Anforderungen mit Serveroptionen

Verwenden Sie in einem System zusammengeschlossener Datenbanken Parameter, die sogenannten *Serveroptionen*, um für den globalen Katalog Informationen bereitzustellen, die für eine gesamte Datenquelle gelten, oder um die Interaktion von DB2 UDB mit einer Datenquelle zu steuern. Beispielsweise können Sie folgende Aktionen ausführen:

- Katalogisieren der Exemplarkennung, indem Sie sie als Wert zur Serveroption NODE hinzufügen.
- Verwenden der Serveroption VARCHAR_NO_TRAILING_BLANKS, um das Optimierungsprogramm zu informieren, dass alle Spalten mit dem Datentyp VARCHAR, die auf dem Datenquellenserver gespeichert sind, keine folgenden Leerzeichen enthalten. Diese Serveroption kann nur für Oracle-Datenquellen verwendet werden. Verwenden Sie diese Option nur dann, wenn Sie sicher sind, dass alle VARCHAR2-Spalten aller Objekte auf dem Server, auf die mit einem Kurznamen verwiesen wird, keine folgenden Leerzeichen enthalten. Geben Sie andernfalls in einer Spaltenoption für die einzelnen Objekte auf dem Server die Spalten an, die keine folgenden Leerzeichen enthalten. Der Name der Spaltenoption lautet ebenfalls VARCHAR_NO_TRAILING_BLANKS.
- Setzen der Serveroption PLAN_HINTS auf einen Wert, der es DB2 ermöglicht, für Oracle-Datenquellen Anweisungsfragmente, die sog. *Planhinweise* (Plan Hints), bereitzustellen. Planhinweise können ein Optimierungsprogramm für Datenquellen bei der Entscheidung unterstützen, welcher Index für den Zugriff auf eine Tabelle und welche Tabellenverknüpfungsfolge für den Abruf von Daten für eine Ergebnismenge verwendet werden sollen.

Normalerweise setzt der Datenbankadministrator Serveroptionen für ein System zusammengeschlossener Datenbanken. Ein Programmierer kann die Serveroptionen, die Unterstützung für die Optimierung von Abfragen bieten, jedoch sinnvoll verwenden. Angenommen, die Serveroption `PLAN_HINTS` ist für die Datenquellen `ORACLE1` und `ORACLE2` auf den Standardwert 'N' (für diese Datenquelle keine Planhinweise verwenden) gesetzt. Nehmen Sie darüber hinaus an, dass Sie eine verteilte Anforderung schreiben, mit der Daten aus `ORACLE1` und `ORACLE2` ausgewählt werden. Sie erwarten, dass Planhinweise die Optimierungsprogramme in diesen Datenquellen dabei unterstützen, ihre Strategien für den Zugriff auf diese Daten zu verbessern. Sie können den Standardwert durch die Einstellung 'Y' (die Planhinweise verwenden) überschreiben, während zwischen Ihrer Anwendung und der zusammengeschlossenen Datenbank eine Verbindung besteht. Wird die Verbindung zu den Datenquellen beendet, wird die Einstellung automatisch auf 'N' zurückgesetzt.

Vorgehensweise:

Verwenden Sie die Anweisung `SET SERVER OPTION`, um Serveroptionen zu setzen oder zu ändern. Um sicherzustellen, dass die Einstellung wirksam wird, müssen Sie die Anweisung `SET SERVER OPTION` sofort nach der Anweisung `CONNECT` angeben. Die Serveroption wird für die Dauer einer Verbindung zur zusammengeschlossenen Datenbank gesetzt.

Empfehlung: Bereiten Sie die Anweisung dynamisch vor. Die Anweisung `SET SERVER OPTION` wirkt sich nur auf dynamische SQL-Anweisungen aus.

Zugehörige Referenzen:

- „`SET SERVER OPTION` statement“ in *SQL Reference, Volume 2*
- Kapitel 21, „Serveroptionen für Systeme zusammengeschlossener Datenbanken“, auf Seite 253

Verwenden von Durchgriffssitzungen in Anwendungen

Durch Durchgriffssitzungen können Anwendungen direkt mit einem Server kommunizieren, indem sie die native Clientzugriffsmethode und die native SQL-Programmversion des Servers verwenden.

Direktes Abfragen von Datenquellen im Durchgriffsmodus

Durchgriffssitzungen werden in folgenden Fällen verwendet:

- Eine Anwendung muss in der Datenquelle ein Objekt erstellen oder eine `INSERT`-, `UPDATE`- oder `DELETE`-Operation ausführen.
- `DB2® UDB` unterstützt keine eindeutige Datenquellenoperation.

Vorgehensweise:

Verwenden Sie die Anweisung `SET PASSTHRU`, um eine Durchgriffssitzung zu starten und um direkt auf einen Server zuzugreifen. Diese Anweisung kann dynamisch abgesetzt werden. Ein Beispiel dieser Anweisung lautet:

```
SET PASSTHRU ORACLE1
```

Diese Anweisung `SET PASSTHRU` öffnet eine Durchgriffssitzung zur Datenquelle unter Verwendung des Servernamens `ORACLE1`. `ORACLE1` ist der Name, den Sie beim Erstellen der Serverdefinition für den Datenquellenserver registriert haben.

Nachdem die Durchgriffssitzung geöffnet wurde, stellen Sie sicher, dass Sie jeweils den richtigen Namen eines Objekts und nicht seinen Kurznamen verwenden, wenn Sie in der Durchgriffssitzung auf Objekte verweisen. Sie müssen die SQL-Programmversion der Datenquelle verwenden, es sei denn, DB2 UDB ist die Datenquelle, auf die verwiesen wird.

Wird in einer Durchgriffssitzung eine statische Anweisung übergeben, wird diese zur Verarbeitung an den Server mit zusammengeschlossenen Datenbanken gesendet. Soll eine SQL-Anweisung zur Verarbeitung an eine Datenquelle übergeben werden, muss die Anweisung in der Durchgriffssitzung dynamisch vorbereitet werden und die Ausführung noch während der Dauer der Sitzung erfolgen. Gehen Sie wie folgt vor, um Anweisungen in einer Durchgriffssitzung dynamisch vorzubereiten:

- Verwenden Sie zum Übergeben der Anweisung SELECT zusätzlich die Anweisung PREPARE, und verwenden Sie dann die Anweisungen OPEN, FETCH und CLOSE, um auf die Ergebnisse Ihrer Abfrage zuzugreifen.
- Es gibt zwei Möglichkeiten, eine andere unterstützte Anweisung als die Anweisung SELECT zu verwenden. Sie können die unterstützte Anweisung mit Hilfe der Anweisung PREPARE vorbereiten und dann mit der Anweisung EXECUTE ausführen. Sie können jedoch auch die Anweisung EXECUTE IMMEDIATE verwenden, um die Anweisung vorzubereiten und auszuführen.

Wenn Sie während einer Durchgriffssitzung den Befehl COMMIT oder ROLLBACK absetzen, wird durch diesen Befehl die aktuelle Arbeitseinheit beendet, nicht jedoch die Durchgriffssitzung.

Überlegungen und Einschränkungen bei der Verwendung des Durchgriffsmodus in Systemen zusammengeschlossener Datenbanken

Bei der Verwendung einer Durchgriffssitzung müssen einige Überlegungen und Einschränkungen beachtet werden. Die folgenden Überlegungen und Einschränkungen gelten für alle Datenquellen:

- Anweisungen, die in einer Durchgriffssitzung vorbereitet werden, müssen in derselben Sitzung ausgeführt werden. Anweisungen, die in einer Durchgriffssitzung vorbereitet, aber außerhalb derselben Sitzung ausgeführt werden, schlagen fehl und führen zum Fehler SQLSTATE 56098.
- Eine Anwendung kann mehrere SET PASSTHRU-Anweisungen ausgeben, es ist jedoch nur die letzte Sitzung aktiv. Wenn eine neue Anweisung SET PASSTHRU aufgerufen wird, beendet sie die vorangegangene Anweisung SET PASSTHRU. Innerhalb einer Durchgriffssitzung können keine Durchgriffe auf mehrere Datenquellen durchgeführt werden.
- Wenn in einer Anwendung mehrere Durchgriffssitzungen verwendet werden, müssen Sie sicherstellen, dass Sie eine COMMIT-Operation durchführen, bevor Sie eine weitere Durchgriffssitzung öffnen. Dadurch wird die Arbeitseinheit für die aktuelle Sitzung abgeschlossen.
- Parametermarken werden in Durchgriffssitzungen nicht unterstützt. Verwenden Sie Hostvariablen an Stelle von Parametermarken.
- Sie können die WITH HOLD-Semantik für einen Cursor verwenden, der in einer Durchgriffssitzung definiert wurde. Sie empfangen allerdings einen Fehler, wenn Sie versuchen, diese Semantik (mit einer COMMIT-Operation) zu verwenden, und die Datenquelle die WITH HOLD-Semantik nicht unterstützt.

- Hostvariablen, die in SQL-Anweisungen in einer Durchgriffssitzung definiert werden, müssen die Form :Hn haben, wobei H in Großbuchstaben vorliegen muss und n eine eindeutige ganze Zahl ist. Die Werte von n müssen beginnend mit Null aufeinander folgend nummeriert werden.
- Der Durchgriff unterstützt keine LOBs.
- Der Durchgriff unterstützt keine Aufrufe gespeicherter Prozeduren.
- Der Durchgriff unterstützt die Anweisung SELECT INTO nicht.

Zugehörige Konzepte:

- „Durchgriffssitzungen“ auf Seite 12
- „Direktes Abfragen von Datenquellen im Durchgriffsmodus“ auf Seite 242

Zugehörige Tasks:

- „Durchgriffssitzungen zu Oracle-Datenquellen“ auf Seite 244
- „Zugreifen auf Datenquellen mit Durchgriffssitzungen“ auf Seite 123

Zugehörige Referenzen:

- „SET PASSTHRU statement“ in *SQL Reference, Volume 2*

Durchgriffssitzungen zu Oracle-Datenquellen

Vor der Übergabe von SQL-Anweisungen an Oracle-Datenquellen in einer Durchgriffssitzung müssen die folgenden Aspekte berücksichtigt werden:

- Wenn ein ferner Client über einen Befehlszeilenprozessor (CLP) im Durchgriffsmodus eine Anweisung SELECT ausgibt und der Client-Code ein SDK vor DB2 Universal Database Version 5 ist, führt diese Anweisung zur Ausgabe einer Nachricht mit dem SQLCODE -30090 und dem Ursachencode 11. Zur Vermeidung dieses Fehlers müssen ferne Clients ein SDK ab Version 5 verwenden.
- Alle DDL-Anweisungen, die auf einem Oracle-Server ausgegeben werden, werden zum Zeitpunkt der syntaktischen Analyse ausgeführt, ohne dass die Transaktionssemantik angewendet wird. Nach dem Abschluss der Operation führt Oracle dafür automatisch eine COMMIT-Operation durch. Wenn eine ROLLBACK-Operation auftritt, wird für die DDL keine ROLLBACK-Operation durchgeführt.
- Wenn Sie für unformatierte Datentypen eine Anweisung SELECT ausgeben, müssen Sie die Funktion RAWTOHEX verwenden, um die Hexadezimalwerte zu empfangen. Wenn Sie für unformatierte Datentypen eine Anweisung INSERT ausführen, müssen Sie die Hexadezimaldarstellung bereitstellen.

Teil 5. Referenz

Kapitel 19. Sichten in der globalen Katalogtabelle mit Informationen der zusammengesetzten Datenbank

Die meisten Katalogsichten in einer zusammengesetzten 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 zusammengesetzter 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 zusammengesetzten Datenbanken enthalten. Auf diese Sichten besteht Lesezugriff.

Tabelle 19. In einem System zusammengesetzter 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 Benutzer und Gruppen.
SYSCAT.FUNCMAPOPTIONS	Enthält Informationen über Optionswerte, die Sie für eine Funktionszuordnung definiert haben.
SYSCAT.FUNCMAPPINGS	Enthält die Funktionszuordnungen zwischen der zusammengesetzten Datenbank und den Datenquellenobjekten.
SYSCAT.INDEXCOLUSE	Enthält Spalten, die an einem Index beteiligt sind.

Tabelle 19. In einem System zusammenschlossener 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 zusammenschlossene 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 zusammenschlossenen Datenbank und den Datenquellenservern eingerichtet werden.
SYSCAT.VIEWS	Enthält Informationen über von Ihnen erstellte lokale Sichten für zusammenschlossene 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 20. Zusammengeschlossene aktualisierbare globale Katalogsichten

Katalogsichten	Beschreibung
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.

Kapitel 20. 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 21. 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 bei Abfragen, die auf Kurznamen verweisen“ auf Seite 181

Zugehörige Tasks:

- „Trusted and fenced mode process environments“ in *IBM DB2 Information Integrator Wrapper Developer's Guide*
- „Ändern eines Wrappers“ auf Seite 31
- „Registrieren von Wrappern für eine Datenquelle“ in *IBM DB2 Information Integrator Konfiguration von Datenquellen*

Kapitel 21. 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 22. 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 22. 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 24 auf Seite 255 aufgelistet.

Tabelle 23. 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 23. Serveroptionen für nicht relationale Datenquellen (Forts.)

Datenquelle	CASE_SENSITIVE	CONTENT_DIR	DAEMON_PORT	ES_HOST	ES_PORT	ES_TRACING	ES_TRACELEVEL	ES_TRACEFILENAME	HMPFAM_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 24. 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 25. 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
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> <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>	Y
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 25. 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 1×10^{23}. 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 25. 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 1×10^{23}. 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 25. 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 25. 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 Tracemeldungen protokolliert. 'ON' Es werden Tracemeldungen 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 25. 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 25. 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/sqlib/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>'W' 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>'N' Setzt den Informix-Sperrmodus auf NOWAIT. Wenn der Wrapper versucht, auf eine gesperrte Tabelle oder Zeile zuzugreifen, gibt Informix einen Fehler zurück.</p> <p>'n' 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 25. 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 1×10^{23}. 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 25. 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 25. 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 25. 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 25. 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 25. 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 25. 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“ auf
- | Seite 158
- | • „Serverkenndaten mit Auswirkungen auf die globale Optimierung“ auf Seite 171

|

| **Zugehörige Tasks:**

- | • „Registrieren von Serverdefinitionen für eine Datenquelle“ in *IBM DB2 Informa-*
- | *tion Integrator Konfiguration von Datenquellen*

|

| **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*

Kapitel 22. 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 26. 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-Domänenamen ein.	Die Standarddomäne zur Authentifizierung für die Documentum-Datenbank.

Tabelle 26. 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"> • Der Konfigurationsparameter AUTHENTICATON des Datenbankmanagers ist auf SERVER gesetzt. • Bei der Herstellung der Verbindung zur DB2-Daten-bank haben Sie eine Berechtigungs-ID und ein zuge-höriges Kennwort angegeben. <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“ in *IBM DB2 Infor-mation Integrator Konfiguration von Datenquellen*

Kapitel 23. Kurznamenoptionen für Systeme zusammengesetzter Datenbanken

Tabelle 27 und Tabelle 28 enthalten eine Liste der Kurznamenoptionen für die verschiedenen Datenquellen. Tabelle 29 auf Seite 274 beschreibt die verschiedenen Kurznamenoptionen und listet die zulässigen sowie die Standardeinstellungen auf.

Tabelle 27. 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 28 enthält eine Auflistung der Kurznamenoptionen (R bis X) der einzelnen Datenquellen.

Tabelle 28. 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 28. 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 29 beschreibt die verschiedenen Kurznamenoptionen und listet die zulässigen sowie die Standardeinstellungen auf.

Tabelle 29. 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 29. 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 29. 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 29. 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 29. 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 29. 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 29. 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 29. 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 29. 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

Kapitel 24. 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 30. 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	XPAT
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 30. 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 31. 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"> • E_value • QueryStrands • GapAlign • NMisMatchPenalty • NMatchReward • Matrix • FilterSequence • NumberOfAlignments • GapCost • ExtendedGapCost • WordSize • ThresholdEx <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 31. 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 31. 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 Kurznamenspaltenoption 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. 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 Rootkurznamenens (d. h. des Kurznamenens 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_PAHT 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>FILE 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>DIRECTORY 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>URI 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>COLUMN Gibt an, dass das XML-Dokument in der Spalte einer relationalen Datenbank gespeichert ist.</p>	

Tabelle 31. 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 Spaltentyp 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 31. 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"> • Sich wiederholende Attribute, deren Typ nicht VARCHAR lautet. • VARCHAR-Spalten, wenn für ALL_VALUES der Wert 'N' angegeben ist. 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 31. 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 31. 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“ auf Seite 156

Zugehörige Tasks:

- „Globale Optimierung“ auf Seite 170

Kapitel 25. 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 32. 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

Kapitel 26. 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 33. 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 34. DB2 für iSeries (und AS/400)

Servertyp	Datenquelle
DB2/400	IBM DB2 für iSeries und AS/400

Tabelle 35. DB2 für z/OS und OS/390

Servertyp	Datenquelle
DB2/ZOS	IBM DB2 für z/OS
DB2/390	IBM DB2 für OS/390

Tabelle 35. DB2 für z/OS und OS/390 (Forts.)

Servertyp	Datenquelle
DB2/MVS	IBM DB2 für MVS

Tabelle 36. 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

Kapitel 27. 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 37. 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 37. 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 38. 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 38. 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 39. 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	-	-	-
I FLOAT	4	-	-	-	-	-	REAL	-	-	-
I 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 40. 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	-
BLOB	-	-	-	-	-	-	BLOB	-	-	-
CHAR	-	-	-	-	-	-	CHAR	-	0	N
CHAR	-	-	-	-	Y	-	CHAR	-	0	Y
CLOB	-	-	-	-	-	-	CLOB	-	-	-
DATE	-	-	-	-	-	-	DATE	-	0	-
DBCLOB	-	-	-	-	-	-	DBCLOB	-	-	-
DECIMAL	-	-	-	-	-	-	DECIMAL	-	-	-
DOUBLE	-	-	-	-	-	-	DOUBLE	-	-	-
FLOAT	-	-	-	-	-	-	DOUBLE	-	-	-
GRAPHIC	-	-	-	-	-	-	GRAPHIC	-	0	N
INTEGER	-	-	-	-	-	-	INTEGER	-	0	-
LONGVAR	-	-	-	-	N	-	CLOB	-	-	-
LONGVAR	-	-	-	-	Y	-	BLOB	-	-	-
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 41. 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 41. 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 42. 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 ⁴	-	-	-	-	-	-	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 42. 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 ¹	1	38	-84	127	-	-	DOUBLE	-	-	-
DUMMY2000 ³	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 ²	-	-	-	-	-	-	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 ⁴	-	-	-	-	-	-	BIGINT	-	-	-
SQL_BINARY	1	254	-	-	-	-	CHARACTER	-	-	Y
SQL_BINARY	255	8000	-	-	-	-	VARCHAR	-	-	Y
SQL_BIT	-	-	-	-	-	-	SMALLINT	2	-	-
SQL_CHAR	1	254	-	-	-	-	CHAR	-	-	N

Tabelle 42. 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 ²	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 ²	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 43. 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 44. 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 ¹	-	-	-	-	-	-	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 45. 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 45. 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 ¹	1	254	-	-	-	-	CHAR	-	-	N
unichar ¹	255	16384	-	-	-	-	VARCHAR	-	-	N
unichar null (siehe univarchar)										
univarchar ¹	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 46. 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 46. 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“ auf Seite 56

Zugehörige Referenzen:

- „Ändern langer Datentypen in VARCHAR-Datentypen“ auf Seite 64
- „Standardwerte für progressive Unicode-Datentypzuordnungen - NET8-Wrapper“ auf Seite 327
- „Standardmäßige progressive Unicode-Datentypzuordnungen - Sybase-Wrapper“ auf Seite 328
- „Standardwerte für progressive Unicode-Datentypzuordnungen - ODBC-Wrapper“ auf Seite 329
- „Standardmäßige progressive Unicode-Datentypzuordnungen - Microsoft SQL Server-Wrapper“ auf Seite 330

Kapitel 28. 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 47. 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 48. 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 49. 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	-	-	-
I 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 50. 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 51. 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 ¹	-	-	-	-	-	-	DECIMAL	19	-	-
BIGINT ²	-	-	-	-	-	-	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 ²	255	2048	-	-	N	-	LVARCHAR	-	-	-
VARCHAR ²	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 52. 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 53. 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 53. 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 54. 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 54. 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 ¹	1	255	-	-	N	-	varchar	-	-	-
VARCHAR ¹	256	32672	-	-	N	-	text	-	-	-
VARCHAR ²	1	16384	-	-	N	-	varchar	-	-	-
VARCHAR ²	16385	32672	-	-	N	-	text	-	-	-
VARCHAR ¹	1	255	-	-	Y	-	varbinary	-	-	-
VARCHAR ¹	256	32672	-	-	Y	-	image	-	-	-
VARCHAR ²	1	16384	-	-	Y	-	varbinary	-	-	-
VARCHAR ²	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 55. 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 ¹	1	64000	-	-	-	-	VARBYTE	-	-	-
CHARACTER	-	-	-	-	-	-	CHARACTER	-	-	-
CHARACTER	-	-	-	-	Y	-	BYTE	-	-	-

| Tabelle 55. 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 ²	1	64000	-	-	-	-	VARCHAR	-	-	-
DATE	-	-	-	-	-	-	DATE	-	-	-
DBCLOB ³	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“ auf Seite 56

Kapitel 29. Standardmäßige Datentypzuordnungen in Unicode

Standardwerte für progressive Unicode-Datentypzuordnungen - NET8-Wrapper

In der folgenden Tabelle sind die Standardwerte für die progressive Datentypzuordnung des NET8-Wrappers aufgelistet, die gelten, wenn als zusammengeslossene Datenbank eine Unicode-Datenbank verwendet wird.

Tabelle 56. Standardwerte für progressive Unicode-Datentypzuordnungen für den NET8-Wrapper

UTF-8		Oracle
Datentyp	Datentyp	Länge
CHAR	CHAR	1 bis 254 Byte
VARCHAR	CHAR	255 bis 2000 Byte
	VARCHAR2	1 bis 4000 Byte
DBCLOB	NCLOB	
GRAPHIC	NCHAR	1 bis 127 Zeichen
VARGRAPHIC	NCHAR	128 bis 1000 Zeichen
	NVARCHAR2	1 bis 2000 Zeichen

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammengeslossener Datenbanken“ auf Seite 145

Standardmäßige regressive Unicode-Datentypzuordnungen - NET8-Wrapper

In der folgenden Tabelle sind die Standardwerte für die regressive Datentypzuordnung des NET8-Wrappers aufgelistet, die gelten, wenn als zusammengeslossene Datenbank eine Unicode-Datenbank verwendet wird.

Tabelle 57. Standardwerte für regressive Unicode-Datentypzuordnungen für den NET8-Wrapper

UTF-8		Oracle
Datentyp	Länge	Datentyp
CHAR	1 bis 254 Byte	CHAR
VARCHAR	1 bis 4000 Byte	VARCHAR2
CLOB	1 bis 2 147 483 647 Byte	CLOB
GRAPHIC	1 bis 127 Zeichen	NCHAR
VARGRAPHIC	1 bis 2000 Zeichen	NVARCHAR2
DBCLOB	1 bis 1 073 741 823 Zeichen	NCLOB

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammenschlossener Datenbanken“ auf Seite 145

Standardmäßige progressive Unicode-Datentypzuordnungen - Sybase-Wrapper

In der folgenden Tabelle sind die Standardwerte für die progressive Datentypzuordnung des CTLIB-Wrappers aufgelistet, die gelten, wenn als zusammenschlossene Datenbank eine Unicode-Datenbank verwendet wird.

Tabelle 58. Standardwerte für progressive Unicode-Datentypzuordnungen für den Sybase CTLIB-Wrapper

UTF-8	Sybase	
Datentyp	Datentyp	Länge
CHAR	char	1 bis 254 Byte
	nchar	1 bis 127 Zeichen
VARCHAR	char	255 bis 32672 Byte
	varchar	1 bis 32672 Byte
	nchar	128 bis 16336 Zeichen
	nvarchar	1 bis 16336 Zeichen
CLOB	text	
GRAPHIC	unichar	1 bis 127 Zeichen
VARGRAPHIC	unichar	128 bis 16336 Zeichen
	univarchar	1 bis 16336 Zeichen

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammenschlossener Datenbanken“ auf Seite 145

Standardmäßige regressive Unicode-Datentypzuordnungen - Sybase-Wrapper

In der folgenden Tabelle sind die Standardwerte für die regressive Datentypzuordnung des CTLIB-Wrappers aufgelistet, die gelten, wenn als zusammengeslossene Datenbank eine Unicode-Datenbank verwendet wird.

Tabelle 59. Standardwerte für regressive Unicode-Datentypzuordnungen für den Sybase CTLIB-Wrapper

UTF-8		Sybase
Datentyp	Länge	Datentyp
CHAR	1 bis 254 Byte	char
VARCHAR	1 bis 32672 Byte	varchar
CLOB	1 bis 2 147 483 647 Byte	text
GRAPHIC	1 bis 127 Zeichen	unichar
VARGRAPHIC	1 bis 16336 Zeichen	univarchar

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammengeslossener Datenbanken“ auf Seite 145

Standardwerte für progressive Unicode-Datentypzuordnungen - ODBC-Wrapper

In der folgenden Tabelle sind die Standardwerte für die progressive Datentypzuordnung des ODBC-Wrappers aufgelistet, die gelten, wenn als zusammengeslossene Datenbank eine Unicode-Datenbank verwendet wird.

Tabelle 60. Standardwerte für progressive Unicode-Datentypzuordnungen für den ODBC-Wrapper

UTF-8		ODBC
Datentyp	Datentyp	Länge
CHAR	SQL_CHAR	1 bis 254 Byte
VARCHAR	SQL_CHAR	255 bis 32672 Byte
	SQL_VARCHAR	1 bis 32672 Byte
CLOB	SQL_LONGVARCHAR	-
GRAPHIC	SQL_WCHAR	1 bis 127 Zeichen
VARGRAPHIC	SQL_WVARCHAR	128 bis 16336 Zeichen
	SQL_WVARCHAR	1 bis 16336 Zeichen
DBCLOB	SQL_WLONGVARCHAR	-

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammengeslossener Datenbanken“ auf Seite 145

Standardmäßige regressive Unicode-Datentypzuordnungen - ODBC-Wrapper

In der folgenden Tabelle sind die Standardwerte für die regressive Datentypzuordnung des ODBC-Wrappers aufgelistet, die gelten, wenn als zusammengeslossene Datenbank eine Unicode-Datenbank verwendet wird.

Tabelle 61. Standardwerte für regressive Unicode-Datentypzuordnungen für den ODBC-Wrapper

UTF-8		ODBC
Datentyp	Länge	Datentyp
CHAR	1 bis 254 Byte	SQL_CHAR
VARCHAR	1 bis 32672 Byte	SQL_VARCHAR
CLOB	1 bis 2 147 483 647 Byte	SQL_LONGVARCHAR
GRAPHIC	1 bis 127 Zeichen	SQL_WCHAR
VARGRAPHIC	1 bis 16336 Zeichen	SQL_WVARCHAR
DBCLOB	1 bis 1 073 741 823 Zeichen	SQL_WLONGVARCHAR

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammengeslossener Datenbanken“ auf Seite 145

Standardmäßige progressive Unicode-Datentypzuordnungen - Microsoft SQL Server-Wrapper

In der folgenden Tabelle sind die Standardwerte für die progressive Datentypzuordnung des Microsoft SQL Server-Wrappers aufgelistet, die gelten, wenn als zusammengeslossene Datenbank eine Unicode-Datenbank verwendet wird.

Tabelle 62. Standardwerte für progressive Unicode-Datentypzuordnungen für den Microsoft SQL Server-Wrapper

UTF-8	Microsoft SQL Server	
Datentyp	Datentyp	Länge
CHAR	CHAR	1 bis 254 Byte
VARCHAR	CHAR	255 bis 8000 Byte
	VARCHAR	1 bis 8000 Byte
CLOB	TEXT	-
GRAPHIC	NCHAR	1 bis 127 Zeichen
VARGRAPHIC	NCHAR	128 bis 16336 Zeichen
	NVARCHAR	1 bis 16336 Zeichen
DBCLOB	NTEXT	-

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammengeslossener Datenbanken“ auf Seite 145

Standardmäßige regressive Unicode-Datentypzuordnungen - Microsoft SQL Server-Wrapper

In der folgenden Tabelle sind die Standardwerte für die regressive Datentypzuordnung des Microsoft SQL Server-Wrappers aufgelistet, die gelten, wenn als zusammengeschlossene Datenbank eine Unicode-Datenbank verwendet wird.

Tabelle 63. Standardwerte für regressive Unicode-Datentypzuordnungen für den Microsoft SQL Server-Wrapper

UTF-8		Microsoft SQL Server
Datentyp	Länge	Datentyp
CHAR	1 bis 254 Byte	CHAR
VARCHAR	1 bis 32672 Byte	VARCHAR
CLOB	1 bis 2 147 483 647 Byte	TEXT
GRAPHIC	1 bis 127 Zeichen	NCHAR
VARGRAPHIC	1 bis 16336 Zeichen	NVARCHAR
DBCLOB	1 bis 1 073 741 823 Zeichen	NTEXT

Zugehörige Konzepte:

- „Unicode-Unterstützung für Systeme zusammengeschlossener Datenbanken“ auf Seite 145

Kapitel 30. Für nicht relationale Datenquellen unterstützte Datentypen

Bei den meisten nicht relationalen Datenquellen müssen Sie die Spalteninformationen einschließlich des Datentyps angeben, wenn Sie die Kurznamen für den Zugriff auf die Datenquelle erstellen.

Einige der nicht relationalen Wrapper erstellen alle erforderlichen Spalten für den Zugriff auf eine Datenquelle. Diese werden als *Spalten mit festgelegtem Format* bezeichnet. Bei anderen Wrappern hingegen können Sie einige oder alle Datentypen für die Spalten in der Anweisung CREATE NICKNAME selbst angeben.

In den folgenden Abschnitten sind die Wrapper aufgelistet, deren Datentypen Sie angeben können. Außerdem finden Sie dort Informationen zu den Datentypen, die vom jeweiligen Wrapper unterstützt werden.

Vom BioRS-Wrapper unterstützte Datentypen

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom BioRS-Wrapper unterstützt werden.

Tabelle 64. BioRS-Datentypen mit Zuordnungen zu DB2-Datentypen

BioRS-Datentypen	DB2-Datentyp
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 beträgt 5 MB.

Vom BLAST-Wrapper unterstützte Datentypen

Einige der Datentypen werden für die Spalten mit festgelegtem Format, die vom BLAST-Wrapper erstellt werden, automatisch definiert.

Für die Definitionszeilenfelder können Sie die Zuordnung beim Erstellen eines Kurznamens durchführen. Wenn die Daten in der Definitionszeilenspalte nicht mit dem lokalen Spaltendatentyp kompatibel sind, wird eine Fehlermeldung ausgegeben. Wenn Sie z. B. eine Definitionszeilenspalte des Typs INTEGER definieren und die Spalte Werte enthält, die nicht numerisch sind, wird eine Fehlermeldung ausgegeben.

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom BLAST-Wrapper unterstützt werden.

Tabelle 65. BLAST-Datentypen mit Zuordnungen zu DB2-Datentypen

BLAST-Datentypen	DB2-Datentyp
definition line	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB beträgt 5 MB.
definition line	DOUBLE
definition line	FLOAT
definition line	INTEGER
definition line	VARCHAR

Vom Documentum-Wrapper unterstützte Datentypen

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom Documentum-Wrapper unterstützt werden.

Tabelle 66. Documentum-Datentypen mit Zuordnungen zu DB2-Datentypen

Documentum-Datentypen	DB2-Datentyp
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

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom Entrez-Wrapper unterstützt werden.

Tabelle 67. Entrez-Datentypen mit Zuordnungen zu DB2-Datentypen

Entrez-Datentypen	DB2-Datentyp
character	CHARACTER
character	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB beträgt 5 MB.
date	DATE
number	DECIMAL
number	DOUBLE
integer	INTEGER
number	REAL
integer	SMALLINT
time	TIMESTAMP
character	VARCHAR

Vom Excel-Wrapper unterstützte Datentypen

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom Excel-Wrapper unterstützt werden.

Tabelle 68. Excel-Datentypen mit Zuordnungen zu DB2-Datentypen

Excel-Datentypen	DB2-Datentyp
date	DATE
number	DOUBLE
number	FLOAT (n); dabei ist n >= 25 und ≤= 53
integer	INTEGER
character	VARCHAR

Vom Extended Search-Wrapper unterstützte Datentypen

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom Extended Search-Wrapper unterstützt werden.

Tabelle 69. Extended Search-Datentypen mit Zuordnungen zu DB2-Datentypen

Extended Search-Datentypen	DB2-Datentyp
Date	DATE
Double	DOUBLE
Integer	INTEGER
String	VARCHAR

Vom HMMER-Wrapper unterstützte Datentypen

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom HMMER-Wrapper unterstützt werden.

Tabelle 70. HMMER-Datentypen mit Zuordnungen zu DB2-Datentypen

HMMER-Datentypen	DB2-Datentyp
character	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB beträgt 5 MB.
character	DOUBLE
character	FLOAT
character	INTEGER
character	VARCHAR

Vom Wrapper für Dateien mit Tabellenstruktur unterstützte Datentypen

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom Wrapper für Dateien mit Tabellenstruktur unterstützt werden.

Tabelle 71. Datentypen für Dateien mit Tabellenstruktur mit Zuordnungen zu DB2-Datentypen

Datentypen für Dateien mit Tabellenstruktur	DB2-Datentyp
character	CHARACTER
character	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB beträgt 5 MB.
number	DECIMAL
number	DOUBLE
number	FLOAT
integer	INTEGER
number	REAL
integer	SMALLINT
character	VARCHAR

Vom Web-Services-Wrapper unterstützte Datentypen

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom Web-Services-Wrapper unterstützt werden. Der Web-Services-Wrapper verwendet XML-Datentypen.

Tabelle 72. XML-Datentypen mit Zuordnungen zu DB2-Datentypen für den Web-Services-Wrapper

XML-Datentypen	DB2-Datentyp
character	CHARACTER
character	CHARACTER FOR BIT DATA
character	CLOB
date	DATE
number	DECIMAL
number	DOUBLE
number	FLOAT
integer	INTEGER
number	REAL
integer	SMALLINT
character	VARCHAR
character	VARCHAR FOR BIT DATA

Vom WebSphere Business Integration-Wrapper unterstützte Datentypen

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom WebSphere Business Integration-Wrapper unterstützt werden. Der WebSphere Business Integration-Wrapper verwendet XML-Datentypen.

Tabelle 73. XML-Datentypen mit Zuordnungen zu DB2-Datentypen für den WebSphere Business Integration-Wrapper

XML-Datentypen	DB2-Datentyp
character	CHARACTER
character	CHARACTER FOR BIT DATA
character	CLOB
date	DATE
number	DECIMAL
number	DOUBLE
number	FLOAT
integer	INTEGER
number	REAL
integer	SMALLINT
character	VARCHAR
character	VARCHAR FOR BIT DATA

Vom XML-Wrapper unterstützte Datentypen

In der folgenden Tabelle werden die DB2-Datentypen aufgelistet, die vom XML-Wrapper unterstützt werden.

Tabelle 74. XML-Datentypen mit Zuordnungen zu DB2-Datentypen für den XML-Wrapper

XML-Datentypen	DB2-Datentyp
character	CHARACTER
character	CHARACTER FOR BIT DATA
character	CLOB
	Die maximal zulässige Länge für den Datentyp CLOB beträgt 5 MB.
date	DATE
number	DECIMAL
number	DOUBLE
number	FLOAT
integer	INTEGER
number	REAL
integer	SMALLINT
character	VARCHAR
character	VARCHAR FOR BIT DATA

Kapitel 31. Systemmonitorelemente der zusammengeschlossenen Datenbanken

Ein System zusammengeschlossener Datenbanken ist ein Mehrdatenbanksystem, das den Fern-Datenzugriff bereitstellt. Es ermöglicht Clientzugriff auf verschiedene Datenquellen, die sich auf verschiedenen, relationalen oder nicht relationalen Plattformen von IBM oder anderen Softwareanbietern befinden können. Das System zusammengeschlossener Datenbanken integriert Zugriff auf verteilte Daten und stellt für Benutzer ein einzelnes Datenbankimage einer heterogenen Umgebung dar.

Die folgenden Elemente listen Informationen dazu auf, wie und wie oft Anwendungen, die in einem System zusammengeschlossener DB2-Datenbanken ausgeführt werden, auf eine Datenquelle zugreifen. Sie listen ebenfalls Informationen dazu auf, wie eine gegebene Anwendung, die auf einem Exemplar eines Servers mit zusammengeschlossenen Datenbanken ausgeführt wird, auf eine Datenquelle zugreift. Zu diesen Elementen gehören folgende:

- `datasource_name` - Data Source Name monitor element
- `disconnects` - Disconnects monitor element
- `insert_sql_stmts` - Inserts monitor element
- `update_sql_stmts` - Updates monitor element
- `delete_sql_stmts` - Deletes monitor element
- `create_nickname` - Create Nicknames monitor element
- `passthru` - Pass-Through monitor element
- `stored_procs` - Stored Procedures monitor element
- `remote_locks` - Remote Locks monitor element
- `sp_rows_selected` - Rows Returned by Stored Procedures monitor element
- `select_time` - Query Response Time monitor element
- `insert_time` - Insert Response Time monitor element
- `update_time` - Update Response Time monitor element
- `delete_time` - Delete Response Time monitor element
- `create_nickname_time` - Create Nickname Response Time monitor element
- `passthru_time` - Pass-Through Time monitor element
- `stored_proc_time` - Stored Procedure Time monitor element
- `remote_lock_time` - Remote Lock Time monitor element

Kapitel 32. Gespeicherte Prozedur SYSPROC.NNSTAT

Abrufen momentan verfügbarer Statistikdaten für einen oder mehrere Kurznamen.

Berechtigung

Keine.

Syntax

```
CALL SYSPROC.NNSTAT(  
    'server/null', 'schema/null', 'nickname/null', 'filepath/null',?,?)
```

Parameterbeschreibungen

server Der Server, auf dem der Server mit zusammengeschlossenen Datenbanken die Kurznamenstatistikdaten erfasst. Dieser Server wird vom Benutzer zum Definieren einer Datenquelle in der zusammengeschlossenen Datenbank registriert. Wenn Sie einen Kurznamen angeben, können Sie für diesen Parameter den Wert NULL definieren.

schema

Wird NULL angegeben, ruft der Server mit zusammengeschlossenen Datenbanken alle Kurznamen ab, die unter dem angegebenen Server definiert sind. Wenn für den Parameter 'server' der Wert NULL angegeben ist, ruft der Server mit zusammengeschlossenen Datenbanken die Statistikdaten des Kurznamens ab, die unter dem angegebenen Schema definiert sind. Wenn der Parameter 'schema' und der Parameter 'nickname' auf NULL gesetzt sind und Sie einen Server angeben, ruft der Server mit zusammengeschlossenen Datenbanken die Statistikdaten vom angegebenen Server ab.

nickname

Der Name des Kurznamens. Wenn Sie einen Kurznamen angeben, dann müssen Sie auch ein Schema definieren.

filepath

Der Pfadname und der Dateiname der Protokolldatei. Der Server mit zusammengeschlossenen Datenbanken erstellt die Protokolldatei auf dem Server. Die im Pfad aufgeführten Verzeichnisse müssen bereits vorhanden sein. Verwenden Sie auf Windows-Systemen zwei Backslashes, um den Protokollpfad anzugeben. Beispiel: c:\\temp\\nnstat.log. Wenn Sie NULL angeben, erstellt der Server mit zusammengeschlossenen Datenbanken kein Protokoll.

Ausgabeparameter

out_SQLCode

Der SQL-Fehler als Ergebnis der Statistikdaten.

out_Trace

Der Trace.

Beispiele

```
CALL SYSPROC.NNSTAT(  
    'NULL', 'ADMIN', 'STAFF', '/home/iuser/reportlogs/log1.txt', ?, ?)  
CALL SYSPROC.NNSTAT(  
    'DB2SERV', 'ADMIN', 'NULL', 'c:\\reports\\log1.txt', ?, ?)  
CALL SYSPROC.NNSTAT(  
    'DB2SERV', 'NULL', 'NULL', 'NULL', ?, ?)
```

Zugehörige Konzepte:

- „Funktion zur Aktualisierung von Kurznamenstatistiken - Übersicht“ auf Seite 219

Zugehörige Tasks:

- „Abrufen von Kurznamenstatistiken“ auf Seite 220

Zugehörige Referenzen:

- „Abrufen von Kurznamenstatistiken von der Befehlszeile aus - Beispiele“ auf Seite 222

Kapitel 33. HADR und Datenquellen in zusammengesetzten Datenbanken

Bei HADR (High Availability Disaster Recovery) sendet eine Primärdatenbank Protokolle an eine Bereitschaftsdatenbank. Wenn die Primärdatenbank ausfällt, können Sie auf die Bereitschaftsdatenbank ausweichen.

Voraussetzungen für relationale Wrapper

Zur Implementierung von HADR für den Einsatz in einer zusammengesetzten Datenbank, die mit relationalen Wrappern arbeitet, müssen die folgenden Bedingungen erfüllt sein:

- Sie müssen auf beiden Systemen identische Partitionsnamen für Datenquellen definieren. Wenn z. B. auf einem System mit der Primärdatenbank eine Partition INF1 definiert ist, die auf ein Informix-Exemplar auf dem Host xyz verweist, muss auf dem System mit der Sekundärdatenbank die Partition INF1 auf dasselbe Informix-Exemplar auf demselben Host verweisen.
- Bei Oracle-Datenquellen muss die Clientversion auf dem System mit der Primärdatenbank mit der Clientversion auf dem System mit der Sekundärdatenbank übereinstimmen. Bei anderen Datenquellen müssen die Clientversionen übereinstimmen.
- Die folgenden Umgebungsvariablen in der Konfigurationsdatei der zusammengesetzten Systeme müssen auf den Systemen mit der Primär- und der Sekundärdatenbank identisch sein:

Oracle: NLS_LANG

Informix: CLIENT_LOCALE, DB_LOCALE, DBNLS

Sybase: SYBASE_CHARSET

Teradata: TERADATA_CHARSET

Voraussetzungen für nicht relationale Wrapper

Bei Wrappern für Flachdateien mit Tabellenstruktur sowie bei XML- und Excel-Wrappern muss auf externe Dateien auf beiden Systemen über dieselben Pfadnamen zugegriffen werden können.

Bei HMMER- und BLAST-Wrappern benötigt jedes System TCP/IP-Zugriff auf den entsprechenden Dämon, wenn dieser nicht auf demselben Computer wie DB2 Information Integrator ausgeführt wird.

Beim Documentum-Wrapper muss sich der Documentum-Client auf dem Sekundärsystem befinden. Die Clientkonfigurationsdateien müssen Verbindungen zum gleichen Docbroker weiterleiten.

Für WebSphere Business Integration:

- Wenn der WebSphere MQ-Manager sich nicht auf demselben System wie der Wrapper befindet, müssen Sie den WebSphere MQ-Client auf dem Sekundärsystem installieren. Das Sekundärsystem muss in der Lage sein, auf denselben WebSphere MQ-Manager zuzugreifen wie das Primärsystem.

- Wenn der WebSphere MQ-Manager und der Wrapper sich auf demselben System befinden, der Adapter jedoch auf einem anderen Computer installiert wurde, muss das Sekundärsystem über einen aktiven WebSphere MQ-Manager verfügen. Wenn das Primärsystem ausfällt, müssen Sie den Adapter abschalten und diesen so rekonfigurieren, dass er auf den neuen WebSphere MQ-Manager verweist. Für den neuen WebSphere MQ-Manager müssen dieselben Warteschlangennamen definiert sein wie für das Primärsystem. Anschließend müssen Sie den Adapter erneut starten, um eine Verbindung zum sekundären WebSphere MQ-Manager herzustellen.
- Wenn der WebSphere MQ-Manager, der Wrapper und der Adapter sich auf demselben System befinden, müssen Sie ein Replikat dieser Konfiguration auf dem Sekundärcomputer erstellen.

Zugehörige Konzepte:

- „HADR (High Availability Disaster Recovery) - Übersicht“ in *Datenwiederherstellung und hohe Verfügbarkeit Handbuch und Referenz*

Kapitel 34. Gateway-Server-Informationen für Steuerkomponententraps abfragen

Um den IBM Kundendienst bei der Diagnose von Ausfällen der DB2 UDB-Steuerkomponente zu unterstützen, werden von DB2 UDB bestimmte Abfrage-Gateway-Server-Informationen in einer Datei aufgezeichnet. Gehen Sie wie folgt vor, um diese Informationen in einer Datei aufzuzeichnen:

- Setzen Sie den Konfigurationsparameter FEDERATED des Datenbankmanagers auf YES.
- Erstellen Sie mindestens ein Serverobjekt.
- Wenn Sie die Hauptspeicherpoolinformationen des Abfrage-Gateways zur selben Datei hinzufügen wollen, müssen Sie die Registervariable DB2MEMDBG auf den Wert ON setzen.

Wenn ein Trap auftritt, wird von der Protokollfunktion eine Nachricht in der Datei db2diag.log aufgezeichnet. Diese Nachricht enthält den vollständigen Pfad der Datei, die die Abfrage-Gateway-Server-Informationen enthält.

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 zusammengeschlossener Datenbanken unter z/OS
- Dokumentation zur Funktion für Systeme zusammengeschlossener 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 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. 10 auf Seite 348 dargestellt.

The screenshot shows the IBM DB2 Information Integrator support website. The top navigation bar includes 'Home', 'Products & services', 'Support & downloads', and 'My account'. The breadcrumb trail reads 'Software > DB2 Information Management > DB2 Information Integration >'. The main heading is 'DB2 Information Integrator'. Below this is a search bar with the prompt 'Enter search terms, phrase, error code or APAR number'. There are three checkboxes for filtering results: 'Solve a problem (FAQs, APARs, Technotes)', 'Download (Fixes, Patches)', and 'Learn (Manual Papers, etc.)'. A 'Submit' button is present. Below the search bar are links for 'Advanced search for this product' and 'Search all software support'. The left sidebar contains a 'Support' section with a list of links: 'My support', 'Submit & track problems', 'How to buy software support', 'Help', 'Site tours', and 'Feedback'. The main content area has three columns: 'Self help' with sub-sections 'Solve a problem' (link to Technotes), 'Download' (links to Get Statistics Utilities and Version 8.1 Fix Packs), and 'Learn' (links to Product information and e-books). The 'Product information' link is highlighted with a red circle and a mouse cursor. The 'Problem submission' section explains that users should submit a problem if they haven't found a solution and have a maintenance contract, with a link to 'Submit & track problem: How to buy support for your software'. There is also an 'Other resources' section.

Abbildung 10. 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 75. DB2 Information Integrator-Dokumentation zur Replikationsfunktion 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	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"> • CD mit der <i>DB2-PDF-Dokumentation</i> • DB2 Information Integrator-Unterstützungswebsite
<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"> • CD mit der <i>DB2-PDF-Dokumentation</i> • 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

Tabelle 75. 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"> In 'DB2 Information - Unterstützung': Produktüberblick > Informationsintegration - Überblick > DB2 Information Integrator - Übersicht > Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation DB2 Information Integrator - Klickstartleiste für die Installation DB2 Information Integrator-Unterstützungswebsite Produkt-CD von <i>DB2 Information Integrator</i>

Dokumentation zur Event-Publishing-Funktion für DB2 Universal Database unter z/OS

Tabelle 76. 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"> CD mit der <i>DB2-PDF-Dokumentation</i> DB2 Information Integrator-Unterstützungswebsite
<i>Replication and Event Publishing Guide and Reference</i>	SC18-7568	<ul style="list-style-type: none"> CD mit der <i>DB2-PDF-Dokumentation</i> DB2 Information Integrator-Unterstützungswebsite
<i>Tuning for Replication and Event Publishing Performance</i>	n/v	DB2 Information Integrator-Unterstützungswebsite

Tabelle 76. 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"> In 'DB2 Information - Unterstützung': Produktüberblick > Informationsintegration - Überblick > DB2 Information Integrator - Übersicht > Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation DB2 Information Integrator - Klickstartleiste für die Installation DB2 Information Integrator-Unterstützungswebsite Produkt-CD von <i>DB2 Information Integrator</i>

Dokumentation zur Event-Publishing-Funktion für IMS und VSAM unter z/OS

Tabelle 77. 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 78. 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"> • CD mit der DB2-PDF-Dokumentation • DB2 Information Integrator-Unterstützungswebsite
<i>Replikation und Event-Publishing - Einführung</i>	GC12-3347	<ul style="list-style-type: none"> • CD mit der DB2-PDF-Dokumentation • 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"> • CD mit der DB2-PDF-Dokumentation • DB2 Information Integrator-Unterstützungswebsite
<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"> • In 'DB2 Information - Unterstützung': Produktüberblick > Informationsintegration - Überblick > DB2 Information Integrator - Übersicht > Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation • DB2 Information Integrator - Klickstartleiste für die Installation • DB2 Information Integrator-Unterstützungswebsite • Produkt-CD von <i>DB2 Information Integrator</i>

Dokumentation zur Funktion für Systeme zusammengeschlossener Datenbanken unter z/OS

Tabelle 79. 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 80. 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"> • CD mit der DB2-PDF-Dokumentation • DB2 Information Integrator-Unterstützungswebsite
<i>C++ API Reference for Developing Wrappers</i>	SC18-9172	<ul style="list-style-type: none"> • CD mit der DB2-PDF-Dokumentation • DB2 Information Integrator-Unterstützungswebsite
<i>Konfiguration von Datenquellen</i>	n/v	<ul style="list-style-type: none"> • CD mit der DB2-PDF-Dokumentation • DB2 Information Integrator-Unterstützungswebsite
<i>Systeme zusammengeschlossener Datenbanken</i>	SC12-3241	<ul style="list-style-type: none"> • CD mit der DB2-PDF-Dokumentation • DB2 Information Integrator-Unterstützungswebsite
<i>Guide to Configuring the Content Connector for VeniceBridge</i>	n/v	DB2 Information Integrator-Unterstützungswebsite

Tabelle 80. DB2 Information Integrator-Dokumentation zur Funktion für Systeme zusammengesetzter 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"> • CD mit der <i>DB2-PDF-Dokumentation</i> • DB2 Information Integrator-Unterstützungswebsite
<i>Java API Reference for Developing Wrappers</i>	SC18-9173	<ul style="list-style-type: none"> • CD mit der <i>DB2-PDF-Dokumentation</i> • DB2 Information Integrator-Unterstützungswebsite
<i>Migration</i>	SC12-3243	<ul style="list-style-type: none"> • CD mit der <i>DB2-PDF-Dokumentation</i> • DB2 Information Integrator-Unterstützungswebsite
<i>Wrapper Developer's Guide</i>	SC18-9174	<ul style="list-style-type: none"> • CD mit der <i>DB2-PDF-Dokumentation</i> • 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"> • In 'DB2 Information - Unterstützung': Produktüberblick > Informationsintegration - Überblick > DB2 Information Integrator - Übersicht > Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation • DB2 Information Integrator - Klickstartleiste für die Installation • DB2 Information Integrator-Unterstützungswebsite • Produkt-CD von <i>DB2 Information Integrator</i>

Dokumentation zu Enterprise Search unter Linux, UNIX und Windows

Tabelle 81. 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 82. DB2 Information Integrator Release-Informationen und Installationsvoraussetzungen

Name	Dateiname	Speicherposition
<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"> • Produkt-CD von <i>DB2 Information Integrator</i> • DB2 Information Integrator - Klickstartleiste für die Installation
<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"> • In 'DB2 Information - Unterstützung': Produktüberblick > Informationsintegration - Überblick > DB2 Information Integrator - Übersicht > Probleme, Fehlerumgehungen und Aktualisierungen der Dokumentation • DB2 Information Integrator - Klickstartleiste für die Installation • DB2 Information Integrator-Unterstützungswebsite • Produkt-CD von <i>DB2 Information Integrator</i>
<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® 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 die Farbe der verwendeten Schriftarten in den DB2-Schnittstellen anpassen. Weitere Informationen hierzu finden Sie unter „Eingabehilfen für Bildschirme“ auf Seite 358.
- DB2-Produkte unterstützen Anwendungen mit Eingabehilfen, die mit der Java™ Accessibility API arbeiten. Weitere Informationen hierzu finden Sie unter „Kompatibilität mit Unterstützungseinrichtungen“ auf Seite 358.
- Die DB2-Dokumentation steht in behindertengerechtem Format zur Verfügung. Weitere Informationen hierzu finden Sie unter „Dokumentation im behindertengerechten Format“ auf Seite 358.

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®-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, der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer gesteuerten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine 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. Die in diesem Handbuch aufgeführten Beispiele sollen lediglich der Veranschaulichung und zu keinem anderen Zweck dienen. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. 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 den IBM Anwendungsprogrammierschnittstellen konform sind.

Kopien oder Teile der Beispielprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (Name Ihrer Firma) (Jahr). Teile des vorliegenden Codes wurden aus Beispielpogrammen der IBM Corp. abgeleitet. © Copyright IBM Corp. *„Jahr/Jahre angeben“*. Alle Rechte vorbehalten.

Marken

Folgende Namen sind in gewissen Ländern Marken der IBM Corporation:

IBM
AIX DataJoiner
DB2
DB2 Connect
DB2 Universal Database
Distributed Relational Database Architecture
DRDA
Informix
iSeries
Lotus
Lotus Notes
MVS
OS/390
VM/ESA
VSE/ESA
WebSphere
z/OS

Folgende Namen sind in gewissen Ländern Marken oder eingetragene Marken anderer Unternehmen:

Java und alle Java-basierten Marken und Logos sind in gewissen Ländern Marken oder eingetragene Marken von 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
 Datenquellen
 einzelne 126
 mehrere ferne 126
 Fragmente 10
 lokale und ferne Datenquellen verbinden 126
 mit Verwendung von Durchgriff 242
Abfrageoptimierung
 Beschreibung 10
ACCOUNTING_STRING, Benutzeroption
 gültige Einstellungen 271
Aktualisierungen
 Berechtigungen 109
 Beschreibung 105
 Einschränkungen 110
 fern 105
 lokal 105
 Referenzielle Integrität 111
 von großen Objekten (LOBs) 112
ALTER NICKNAME, Anweisung
 Beispiel
 lokale Spaltennamen ändern 43
 Spaltenoptionen ändern 45
 Beschreibung 116
 Einschränkungen 41
ALTER NICKNAME, Anweisung
 Beispiel
 Lokaler Datentyp 63
ALTER SERVER, Anweisungen 32
 in zusammengeschlossenen Datenbanken, Beispiel 34, 35
ALTER TABLE, Anweisung
 in Systemen zusammengeschlossener Datenbanken, Beschreibung 116
ALTER USER MAPPING, Anweisung
 in zusammengeschlossenen Datenbanken, Beispiel 37
ALTER WRAPPER, Anweisung 31, 32
ändern 32
 Kurznamen
 Überblick 39
 lange Datentypen 64
Ändern
 Kurznamen
 Kurznamenoptionen 44
 Lokaler Datentyp 61
Anwendungen
 auf Datenquellenobjekte verweisen 230
 Datenquelleninformationen katalogisieren 232
 Isolationsstufen 235
 Kurznamen in 229
 Serveroptionen definieren 241
 Szenario für Systeme zusammengeschlossener Datenbanken 227
 verteilte Anforderungen 239
Anwendungsprogramme 80

Archivprotokollierung (siehe Protokollierung mit Protokollspeicherung) 207
Auslöser
 für Kurznamen 114
Autarkie
 in Anweisungen gewährleisten 112

B

Befehlszeilenprozessor (Command Line Processor, CLP)
 Funktionen für Systeme zusammengeschlossener Datenbanken 25
Befehlszentrale
 für Systeme zusammengeschlossener Datenbanken einsetzen 25
Beispiele 136
Benutzerdefinierte Funktionen (UDFs, User-Defined Functions) 21
 in Anwendungen in Systemen zusammengeschlossener Datenbanken 80
 Transaktionsunterstützung für 105
Benutzerdefinierte Typen (User-Defined Types, UDTs)
 nicht unterstützte Datentypen 21
Benutzerzuordnungen
 ändern 37
 Beschreibung 17
 gültige Einstellungen 271
 löschen 50
 Optionen 17
BioRS
 unterstützte Datentypen 333
BLAST
 gültige Objekte für Kurznamen 19
 unterstützte Datentypen 333
 unterstützte Versionen 6

C

Cachetabellen 204
 Assistent 207
 Beschreibung 205
 Cache aktivieren 208
 erstellen 207
 ferne Datenbanken (Katalog) 207
 gespeicherte Abfragetabellen hinzufügen 209
 gespeicherte Abfragetabellen
 löschen 210
 Komponenten 205
 löschen 210
Caching 199, 200, 201
 Cachetabellen erstellen 207
 Cachetabellen löschen 210
CLP (Command Line Processor = Befehlszeilenprozessor)
 Funktionen für Systeme zusammengeschlossener Datenbanken 25
CODEPAGE, Option 148

Codepages 145, 148, 149, 150
 Beschreibung 23
COLLATING_SEQUENCE, Serveroption
 Auswirkungen auf Pushdown-Möglichkeiten 158
 Beispiel 23
 globale Optimierung, Auswirkungen 171
 gültige Einstellungen 253
COMM_RATE, Serveroption
 globale Optimierung, Auswirkungen 171
 gültige Einstellungen 253
COMMENT ON, Anweisung 231
 in Systemen zusammengeschlossener Datenbanken, Beschreibung 116
COMMIT, Anweisung
 Durchgriff 242, 243
CONNECTSTRING, Serveroption
 gültige Einstellungen 253
CPU_RATIO, Serveroption
 globale Optimierung, Auswirkungen 171
 gültige Einstellungen 253
CREATE ALIAS, Anweisung
 in Systemen zusammengeschlossener Datenbanken, Beschreibung 116
CREATE FUNCTION, Anweisung
 (Quellenfunktion oder Schablone) 69, 70, 71
CREATE FUNCTION MAPPING, Anweisung 67, 69, 73, 76, 77, 78, 79, 81
 Funktionsnamen angeben 75
CREATE INDEX, Anweisung 22, 83, 84, 86, 88
 in Systemen zusammengeschlossener Datenbanken, Beschreibung 116
CREATE NICKNAME, Anweisung 56, 121, 126
CREATE SERVER, Anweisung 4
CREATE TABLE, Anweisung
 in Systemen zusammengeschlossener Datenbanken, Beschreibung 116
CREATE TYPE MAPPING, Anweisung 55, 57, 58, 59, 60
CREATE VIEW, Anweisung 234
 in Systemen zusammengeschlossener Datenbanken, Beschreibung 116
Cursorstabilität (CS)
 Isolationsstufe 235

D

DATALINK, Datentyp
 nicht unterstützt 21
DATEFORMAT, Serveroption
 gültige Einstellungen 253
Dateien mit Tabellenstruktur
 gültige Objekte für Kurznamen 19
 Unicode-Unterstützung 149
 unterstützte Datentypen 333

- Daten mit Tabellenstruktur (*Forts.*)
 - unterstützte Versionen 6
 - Datenquellen 8, 10
 - Beschreibung 5
 - Daten aktualisieren 131
 - Daten einfügen 130
 - Daten löschen 132
 - Durchgriff für Abfrage verwenden 242
 - E/A-Geschwindigkeit und Leistung 171
 - einzelne Datenquelle abfragen 126
 - ferne Planhinweise und Leistung 171
 - gültige Servertypen 293
 - lokale Datenquelle mit ferner Datenquelle verknüpfen 126
 - mehrere ferne Datenquellen abfragen 126
 - mit Durchgriff zugreifen auf 123
 - Prozessorgeschwindigkeit und Leistung 171
 - Sortierfolge und Leistung 171
 - Standardnamen für Wrapper 15
 - Übertragungsgeschwindigkeit und Leistung 171
 - Zugriff durch Sichten für zusammengesetzte Datenbanken 124
 - Datenquellen konfigurieren
 - Kurznamenoptionen 273
 - Datenquellenobjekte
 - Beschreibung 17
 - gültige Objekttypen 19
 - Operationen ausführen 231
 - Datentypen 95
 - Auswirkungen auf Pushdown-Analyse 163
 - für nicht relationale Datenquellen 333
 - nicht unterstützt 21
 - Datentypen für große Objekte (LOB)
 - Aktualisierungsoperationen 112
 - Einschränkungen 239
 - Querverweise 239
 - Datentypzuordnungen
 - Auswirkungen auf Pushdown-Analyse 158
 - Beschreibung 21
 - für ein bestimmtes Datenquellenobjekt 61, 63
 - für einen bestimmten Datenquellentyp 58
 - für einen bestimmten Server 60
 - für einen bestimmten Typ und eine bestimmte Version eines Servers 59
 - geeigneter Zeitpunkt für Erstellung 53
 - Hinweise zur Erstellung 57
 - in einem System zusammengesetzter Datenbanken 54
 - nicht relationale 56
 - nicht unterstützte Datentypen 53
 - progressive 299
 - Beschreibung 56
 - regressive 315
 - Beschreibung 56
 - Situationen, in denen neue Zuordnungen erforderlich sind 55
 - Datentypzuordnungen (*Forts.*)
 - Syntax 56
 - DB2 für iSeries
 - gültige Objekte für Kurznamen 19
 - gültige Servertypen 293
 - Isolationsstufen 235
 - LOB-Unterstützung in Systemen zusammengesetzter Datenbanken 237
 - standardmäßige progressive Typzuordnungen 299
 - standardmäßige regressive Typzuordnungen 315
 - Standardname für Wrapper 15
 - unterstützte Versionen 6
 - DB2 für Linux, UNIX und Windows
 - gültige Objekte für Kurznamen 19
 - gültige Servertypen 293
 - Isolationsstufen 235
 - LOB-Unterstützung in Systemen zusammengesetzter Datenbanken 237
 - standardmäßige progressive Typzuordnungen 299
 - standardmäßige regressive Typzuordnungen 315
 - Standardname für Wrapper 15
 - unterstützte Versionen 6
 - DB2 für VM und VSE
 - gültige Objekte für Kurznamen 19
 - gültige Servertypen 293
 - Isolationsstufen 235
 - LOB-Unterstützung in Systemen zusammengesetzter Datenbanken 237
 - standardmäßige progressive Typzuordnungen 299
 - standardmäßige regressive Typzuordnungen 315
 - Standardname für Wrapper 15
 - unterstützte Versionen 6
 - DB2 für z/OS und OS/390
 - gültige Objekte für Kurznamen 19
 - gültige Servertypen 293
 - Isolationsstufen 235
 - LOB-Unterstützung in Systemen zusammengesetzter Datenbanken 237
 - standardmäßige progressive Typzuordnungen 299
 - standardmäßige regressive Typzuordnungen 315
 - Standardname für Wrapper 15
 - unterstützte Versionen 6
 - DB2_MAXIMAL_PUSHDOWN, Serveroption
 - Auswirkungen auf Pushdown-Möglichkeiten 158
 - Entscheidungen für Pushdown-Analyse 165
 - gültige Einstellungen 253
 - db2exfmt, Tool
 - Zugriffspläne anzeigen 165, 176
 - db2expln, Tool
 - Zugriffspläne anzeigen 165, 176
 - DBNAME, Serveroption
 - gültige Einstellungen 253
 - DELETE, Anweisung 110
 - Entscheidungen für die Auswertung von Zugriffsplänen 167
 - in Systemen zusammengesetzter Datenbanken, Beispiele 132
 - in Systemen zusammengesetzter Datenbanken, Beschreibung 116
 - Diagnose
 - in Systemen zusammengesetzter Datenbanken 345
 - Diagnoseanzeiger
 - in Systemen zusammengesetzter Datenbanken 139
 - Diagnosezentrale
 - Diagnoseanzeiger 137
 - Diagnoseanzeiger für Kurznamen und Server für zusammengesetzte Datenbanken konfigurieren 139
 - Diagnosemomentaufnahme 140
 - Status von Kurznamen und Servern zusammengesetzter Datenbanken 139
 - Direktaufruf über die Tastatur
 - Unterstützung bei 357
 - DISABLE, Funktionszuordnungsoption
 - gültige Einstellungen 291
 - Documentum
 - gültige Objekte für Kurznamen 19
 - unterstützte Datentypen 333
 - unterstützte Versionen 6
 - DROP, Anweisung 82
 - Benutzerzuordnungen 50
 - in Systemen zusammengesetzter Datenbanken, Beschreibung 116
 - Kurznamen 51
 - Serverdefinitionen 49
 - Wrapper 47
 - Durchgriff
 - Beschreibung 12, 123
 - COMMIT, Anweisung 242, 243
 - Einschränkungen 12
 - LOB-Unterstützung 239
 - SET PASSTHRU, Anweisung 243
 - SET PASSTHRU RESET, Anweisung 243
 - SQL-Verarbeitung 242
 - Transaktionsunterstützung für 105
 - Überlegungen, Einschränkungen 243
 - dynexpln, Tool
 - Zugriffspläne anzeigen 165, 176
- ## E
- Eingabehilfen
 - Funktionen 357
 - Einphasige Festschreibungsoperationen
 - Definition 103
 - Einschränkungen
 - Kurznamen ändern 41
 - Entrez
 - gültige Objekte für Kurznamen 19
 - unterstützte Versionen 6
 - Excel-Dateien
 - gültige Objekte für Kurznamen 19
 - unterstützte Datentypen 333
 - unterstützte Versionen 6
 - EXPLAIN-Einrichtung 190

EXPLAIN-Einrichtung (*Forts.*)
Kurznamen 189
Explain-Tools 196
Extended Search
gültige Objekte für Kurznamen 19
unterstützte Datentypen 333
unterstützte Versionen 6

F

Fehlerbehebung 345
Fernre Aktualisierungen 105
Fernre Kataloginformationen 9
Fernre Objekte
Beispiele 114
Beschreibung 114
Fernre SQL-Generierung 170
Fernre Tabellen
ändern 99
erstellen 95
löschen 101
Siehe auch Transparente DDL 93
Flachdateien
siehe auch Dateien mit Tabellenstruktur 6
FOLD_ID, Serveroption
Beispiel 35
gültige Einstellungen 253
FOLD_PW, Serveroption
Beispiel 35
gültige Einstellungen 253
Funktionsschablonen
Beschreibung 70
erstellen 71
Funktionszuordnung 75
Funktionszuordnungen
Auswirkungen auf Pushdown-Analyse 158
Beschreibung 21, 69
erstellen 76
bestimmter Datenquellen-server 79
bestimmter Datenquellentyp 77
bestimmter Datenquellentyp und bestimmte Version 78
löschen 82
Optionen
gültige Einstellungen 291
Systemaufwand der Funktion 73
Standardzuordnungen 67
Standardzuordnungen inaktivieren 81
Zuordnung zu UDFs 70

G

Gemischte Parallelität
Datenquellen in zusammengeschlossenen Datenbanken
aktivieren 189
Datenverarbeitung 188
Überblick 181
Zugriffsplan 196
Geschäftsanwendungen
unterstützte Datentypen 333

Gespeicherte Abfragetabellen (MQTs) 200, 201
Cache aktivieren 208
Cachetabellenkomponente 205
Einschränkungen für Kurznamen 204
für Kurznamen 163
hinzufügen zu einer Cachetabelle 209
in Systemen zusammengeschlossener Datenbanken
Überblick 199
löschen aus einer Cachetabelle 210
Gespeicherte Prozeduren
Kurznamen 233
Kurznamenstatistiken 341
Globale Optimierung 169
Kurznamenkenndaten, Auswirkungen 173
Serverkenndaten, Auswirkungen 171
Überblick 170
Globaler Katalog 54
Beschreibung 9
Sichten mit Informationen der zusammengeschlossenen Datenbanken 247
Statistiken aktualisieren 169, 232
GRANT, Anweisung
in Systemen zusammengeschlossener Datenbanken, Beschreibung 116
Kurznamen 231
GROUP BY, Operator
Entscheidungen für die Auswertung von Zugriffsplänen 167
Entscheidungen für die Optimierung von Zugriffsplänen 178
Gruppenoperatoren
Beispiel für verteilte Anforderung 239
Entscheidungen für die Auswertung von Zugriffsplänen 167

H

HADR (High Availability Disaster Recovery)
in Systemen zusammengeschlossener Datenbanken 343
HMMER-Datenquelle
gültige Objekte für Kurznamen 19
unterstützte Datentypen 333
unterstützte Versionen 6

I

IFILE, Serveroption
gültige Einstellungen 253
IGNORE_UDT, Serveroption
gültige Einstellungen 253
Indizespezifikationen
Beschreibung 22
für Datenquellenobjekte 84
für Informix-Synonyme 89
für Sichten 88
für Tabellen mit neu hinzugefügten Indizes 86

Indizespezifikationen (*Forts.*)
globale Optimierung, Auswirkungen 173
in Systemen zusammengeschlossener Datenbanken 83
Informative Integritätsbedingungen
Kurznamen 213, 215

Informix

gültige Objekte für Kurznamen 19
gültige Servertypen 293
Isolationsstufen 235
LOB-Unterstützung in Systemen zusammengeschlossener Datenbanken 237
standardmäßige progressive Typzuordnungen 299
standardmäßige regressive Typzuordnungen 315
Standardname für Wrapper 15
unterstützte Versionen 6
INFORMIX_LOCK_MODE, Serveroption
gültige Einstellungen 253
INITIAL_INSTS, Funktionszuordnungsoption
gültige Einstellungen 291
INITIAL_IOS, Funktionszuordnungsoption
gültige Einstellungen 291
INSERT, Anweisung 109, 110
Entscheidungen für die Auswertung von Zugriffsplänen 167
in Systemen zusammengeschlossener Datenbanken, Beispiele 130
in Systemen zusammengeschlossener Datenbanken, Beschreibung 116
INSTS_PER_ARGBYTE, Funktionszuordnungsoption
gültige Einstellungen 291
INSTS_PER_INVOC, Funktionszuordnungsoption
gültige Einstellungen 291
Integrierte Funktionen 21
IO_RATIO, Serveroption
globale Optimierung, Auswirkungen 171
gültige Einstellungen 253
IOS_PER_ARGBYTE, Funktionszuordnungsoption
gültige Einstellungen 291
IOS_PER_INVOC, Funktionszuordnungsoption
gültige Einstellungen 291
Isolationsstufen 235
IUD_APP_SVPT_ENFORCE, Serveroption
Beispiele 112
gültige Einstellungen 253

K

Katalog
Siehe Globaler Katalog 247
Katalogstatistik
globale Optimierung, Auswirkungen 173
Kompensation, Beschreibung 11
Körperliche Beeinträchtigung 357

Kurznamen
 ändern
 Einschränkungen 41
 Kurznamenoptionen 44
 Lokale Spaltennamen 24
 Lokaler Datentyp 61
 lokaler Datentyp, Beispiel 63
 Spaltenoptionen 45
 Überblick 39
 Beschreibung 17
 erstellen
 Datenquellenobjekte 121
 für Kurznamen 126
 gespeicherte Prozeduren 233
 gültige Datenquellenobjekte 19
 in SQL-Anweisungen 116
 in SQL-Anweisungen verweisen
 auf 230
 Integritätsbedingungen 111
 löschen 51
 Spaltenoptionen definieren 233
 zugreifen auf Datenquellen 229
 Kurznamenoptionen
 angeben 44
 Kurznamenspaltenoptionen
 Beispiele 45
 Beschreibung 20
 Kurznamenstatistiken
 aktualisieren 220
 aktualisieren, Voraussetzung 223
 Aktualisierungsstatus anzeigen 224

L

Leistung 156, 165, 183, 186, 187, 188,
 189, 213, 224
 CPU-Geschwindigkeit 171
 E/A-Geschwindigkeit 171
 ferne Planhinweise 171
 in Systemen zusammengeschlossener
 Datenbanken 153, 213, 215, 219,
 222, 341
 Indexspezifikationen 173
 Katalogstatistik 173
 Siehe auch Optimierung 153
 Sortierfolge 171
 Sortierfolgen 158
 SQL-Unterschiede 158
 Übertragungsgeschwindigkeit 171
 Lesestabilität (RS)
 Isolationsstufe 235
 LOB-Datentypen (Large Object)
 Aktualisierungsoperationen 112
 Einschränkungen 239
 Querverweise 239
 LOCK TABLE, Anweisung
 in Systemen zusammengeschlossener
 Datenbanken, Beschreibung 116
 LOGIN_TIMEOUT, Serveroption
 gültige Einstellungen 253
 Lokale Aktualisierungen 105
 Lokale Objekte
 Beschreibung 114
 Lokaler Katalog
 Siehe Globaler Katalog 9
 LONG-Datentypen 64

M

Microsoft Excel
 Siehe Excel-Dateien 6
 Microsoft SQL Server
 gültige Objekte für Kurznamen 19
 gültige Servertypen 293
 Isolationsstufen 235
 LOB-Unterstützung in Systemen
 zusammengeschlossener Datenban-
 ken 237
 standardmäßige progressive Typzu-
 ordnungen 299
 standardmäßige regressive Typzu-
 ordnungen 315
 Standardnamen für Wrapper 15
 Unicode-Unterstützung 147, 148
 unterstützte Versionen 6
 MODULE, Wrapperoption
 gültige Einstellungen 251
 Monitorschalter
 in Systemen zusammengeschlossener
 Datenbanken 179
 MQTs (gespeicherte Abfragetabellen)
 Cache aktivieren 208
 Cachetabellenkomponente 205
 Einschränkungen für Kurz-
 namen 204
 für Kurznamen 163
 hinzufügen zu einer Cache-
 tabelle 209
 in Systemen zusammengeschlossener
 Datenbanken
 Überblick 199
 löschen aus einer Cachetabelle 210

N

Nicht festgeschriebener Lesevorgang (UR)
 Isolationsstufen 235
 Nicht relationale Datenquellen
 Datentypzuordnungen angeben 21
 unterstützte Datentypen 333
 NODE, Serveroption, gültige Einstellun-
 gen 253
 NUMERIC_STRING, Spaltenoption
 Auswirkungen auf Pushdown-Mög-
 lichkeiten 163
 Beispiel 45
 gültige Einstellungen 283

O

ODBC
 gültige Objekte für Kurznamen 19
 gültige Servertypen 293
 Isolationsstufen 235
 LOB-Unterstützung in Systemen
 zusammengeschlossener Datenban-
 ken 237
 standardmäßige progressive Typzu-
 ordnungen 299
 Standardname für Wrapper 15
 Unicode-Unterstützung 147, 148
 unterstützte Versionen 6
 OLE DB
 gültige Servertypen 293

OLE DB (Forts.)

Isolationsstufen 235
 Standardname für Wrapper 15
 unterstützte Versionen 6
 Optimierung
 Abfrageverarbeitung 153
 Gespeicherte Abfragetabellen 163
 Indexspezifikationen 171
 Katalogstatistik 171
 Kurznamenspaltenoptionen 163
 Serverkennndaten, Auswirkungen 171
 Serveroptionen 158
 Siehe auch Leistung 153
 Sortierfolgen 158
 verteilte Anforderungen 241
 Optimierungsprogramm
 Beschreibung 10
 Fixkostenmodell 170
 Optionen
 Kurznamen 273
 Oracle
 gültige Objekte für Kurznamen 19
 Isolationsstufen 235
 LOB-Unterstützung in Systemen
 zusammengeschlossener Datenban-
 ken 237
 standardmäßige progressive Typzu-
 ordnungen 299
 standardmäßige regressive Typzu-
 ordnungen 315
 Standardnamen für Wrapper 15
 ORDER BY, Operator
 Entscheidungen für die Auswertung
 von Zugriffsplänen 167

P

PACKET_SIZE, Serveroption
 gültige Einstellungen 253
 Parallelität 182, 183, 186, 188, 189
 in Systemen zusammengeschlossener
 Datenbanken 181, 183
 Partitionsinterne Parallelität 181
 in Systemen zusammengeschlossener
 Datenbanken 182, 183
 Zugriffspläne für Systeme zusammen-
 geschlossener Datenbanken 190
 Partitionsübergreifende Parallelität 181,
 187
 in Systemen zusammengeschlossener
 Datenbanken 183, 186, 188, 192
 PASSWORD, Serveroption
 gültige Einstellungen 253
 PERCENT_ARGBYTES, Funktionszu-
 ordnungsoption
 gültige Einstellungen 291
 PLAN_HINTS, Serveroption
 Beispiel 35
 globale Optimierung, Auswirkun-
 gen 171
 gültige Einstellungen 253
 Progressive Typzuordnungen
 Beschreibung 56
 Standardzuordnungen 299
 Unicode 327, 328, 329, 330
 Protokollierung mit Protokollspeicherung
 Beschreibung, Cachetabellen 207

PUSHDOWN, Serveroption
gültige Einstellungen 253
Pushdown-Analyse
Abfragekenndaten, Auswirkungen 165
Beschreibung 10, 156
Kurznamenkenndaten, Auswirkungen 163
Serverkenndaten, Auswirkungen 158

R

RAWTOHEX, Funktion 245
Rechenpartitionsgruppen 186, 187
Referenzielle Integrität 111
Regeln
Zuordnungssemantik in einem System
zusammengeschlossener Datenbanken 133
Regressive Typzuordnung
Unicode 327
Regressive Typzuordnungen
Beschreibung 56
Standardzuordnungen 315
Unicode 329, 330, 331
REMOTE_AUTHID, Benutzeroption
Beispiel 37
gültige Einstellungen 271
REMOTE_DOMAIN, Benutzeroption
gültige Einstellungen 271
REMOTE_NAME, Funktionszuordnungsoption
gültige Einstellungen 291
REMOTE_PASSWORD, Benutzeroption
Beispiel 37
gültige Einstellungen 271
REVOKE, Anweisung
in Systemen zusammengeschnittener
Datenbanken, Beschreibung 116
Rohdatentypen
SQL-Anweisung, Übergabe 245

S

Schreiboperationen
Siehe Aktualisierungen 105
SELECT, Anweisung
in Systemen zusammengeschnittener
Datenbanken, Beispiele 126
in Systemen zusammengeschnittener
Datenbanken, Beschreibung 116
Server mit zusammengeschnittenen
Datenbanken 5
Beschreibung 4
Wrapper 13
Wrappermodule 13
Serverdefinitionen
alle Datenquellendefinitionen
ändern 35
ändern, Übersicht und Einschränkungen 32
Beschreibung 16
Datenquellenversion ändern 34
löschen 49
Serveroptionen 35

Serveroptionen
Auswirkungen auf Pushdown-Analyse 158
Beschreibung 16
globale Optimierung, Auswirkungen 171
gültige Einstellungen 253
Hierarchiesatz 35
hinzufügen und ändern 35
temporär festlegen 35
temporäre 16
verteilte Anforderungen optimieren 241
Servertypen
gültige Typen für Server mit
zusammengeschnittenen Datenbanken 293
SET PASSTHRU, Anweisung
Überlegungen 243
SET SERVER OPTION, Anweisung
Beispiel 35
Option temporär festlegen 16
verteilte Anforderungen optimieren 241
Sicherung (Backup) 343
Sicherungspunkte
Datenquellen-APIs 112
Sichten für zusammengeschnittene
Datenbanken
Beispiele 124, 234
erstellen 124, 234
Snapshot Monitor 141
Fragmente von Abfragen für
zusammengeschnittene Datenbanken 142
Kurznamen und Server 140
Kurznamen und Server
zusammengeschnittener Datenbanken 137, 139
Sortierfolgen
Beschreibung 23
planen 23
Überblick 158
Sortierung 23
Spaltennamen
ändern 43
Spaltenoptionen
Auswirkungen auf Pushdown-Analyse 163
Beschreibung 20
definieren 233
für Kurznamen angeben 45
gültige Einstellungen 283
NUMERIC_STRING 233
VARCHAR_NO_TRAILING-
_BLANKS 233
SQL-Anweisungen
Kurznamenunterstützung 114, 116
SQL-Compiler
Flussdiagramm einer Abfrage-
analyse 153
in einem System zusammengeschnittener
Datenbanken 10
SQL Explain
Zugriffspläne anzeigen 165, 176
SQL-Programmversion
Auswirkungen auf Pushdown-Analyse 158
Beschreibung 11

Statistikdaten
Kurzname 219, 222
Statistiken für Systeme
zusammengeschnittener Datenbanken
aktualisieren 219
Steuerzentrale
Schnittstelle für Systeme
zusammengeschnittener Datenbanken 25
Sybase
gültige Objekte für Kurznamen 19
gültige Servertypen 293
Isolationsstufen 235
LOB-Unterstützung in Systemen
zusammengeschnittener Datenbanken 237
standardmäßige progressive Typzuordnungen 299
standardmäßige regressive Typzuordnungen 315
Standardnamen für Wrapper 15
unterstützte Versionen 6
Synonyme
Indizespezifikationen für Informix
erstellen 89
SYSCAT, Katalogsichten 70, 247
SYSCAT.TABLES 232
SYSPROC.NNSTAT, gespeicherte
Prozedur 220
SYSSTAT, Katalogsichten 247
Systeme zusammengeschnittener
Datenbanken
Überblick 3
Systemmonitorschalter
in Systemen zusammengeschnittener
Datenbanken 179
Szenarios 227

T

Teradata
gültige Objekte für Kurznamen 19
gültige Servertypen 293
Isolationsstufen 235
LOB-Unterstützung in Systemen
zusammengeschnittener Datenbanken 237
standardmäßige progressive Typzuordnungen 299
standardmäßige regressive Typzuordnungen 315
Standardname für Wrapper 15
TIMEFORMAT, Serveroption
gültige Einstellungen 253
TIMEOUT, Serveroption
Beispiel 35
gültige Einstellungen 253
TIMESTAMPFORMAT, Serveroption
gültige Einstellungen 253
Toolskatalog
Datenbank erstellen 223
Transaktionen
Aktualisierungen 105
Überblick 103
Transparente DDL
Beschreibung 93
ferne Tabellen ändern 99
ferne Tabellen erstellen 95

Transparente DDL (*Forts.*)
ferne Tabellen löschen 101
Spaltenlängen für LOB 95
Transaktionsunterstützung für 105

U

Unicode 145, 147, 148, 149, 150, 327, 328,
329, 330, 331
Unterabfragen
Beispiel für verteilte Anforderung 239
UPDATE, Anweisung 110
Entscheidungen für die Auswertung
von Zugriffsplänen 167
in Systemen zusammengeschlossener
Datenbanken, Beispiele 131
in Systemen zusammengeschlossener
Datenbanken, Beschreibung 116

V

VARCHAR_NO_TRAILING_BLANKS,
Serveroption
Auswirkungen auf Pushdown-Möglich-
lichkeiten 158
gültige Einstellungen 253
VARCHAR_NO_TRAILING_BLANKS,
Spaltenoption
Auswirkungen auf Pushdown-Mög-
lichkeiten 163
Beispiel 45
gültige Einstellungen 283
Vergleichselemente
Entscheidungen für die Auswertung
von Zugriffsplänen 167
Verknüpfungen
Beispiel für verteilte Anforderung 239
Entscheidungen für die Optimierung
von Zugriffsplänen 178
Verteilte Anforderungen
codieren 239
optimieren 241
Verteiltes Datenbankmanagement-
system 3
Visual Explain
Zugriffspläne anzeigen 165, 176

W

Web-Services
unterstützte Datentypen 333
WebSphere
Szenario 227
WebSphere Business Integration-Wrapper
unterstützte Datentypen 333
Wiederherstellung
HADR und Datenquellen in zusam-
mengeschlossenen Datenban-
ken 343
Wiederholtes Lesen (RR)
Isolationsstufe 235
WITH HOLD, Cursorsemantik
für einen Kurznamen 114
in Durchgriffssitzungen 123

Wrapper
ändern 31
Beschreibung 13
löschen 47
standardmäßige Namen 15
Wrapperoptionen
gültige Einstellungen 251

X

XML
gültige Objekte für Kurznamen 19
unterstützte Datentypen 333
unterstützte Versionen 6
XML-Wrapper 153

Z

Zeichenfolgen
Sortierfolgen 23
Zeichensätze
Beschreibung 23
Zeitmarkenmonitorschalter 179
Zugriffsplan 189, 196
Zugriffspläne 192
anzeigen 165, 176
Beschreibung 10
Entscheidungen für Auswertung 167
Entscheidungen für Optimierung 178
Leistung 178
Zuordnungen
in Systemen zusammengeschlossener
Datenbanken 133
Zuordnungssemantik in einem System
zusammengeschlossener Datenbanken
Beispiele 136
Zusammengeschlossene Datenbanken
Beschreibung 8
Systemkatalog 9
Zweiphasige Festschreibung
Operationen 103

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 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 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 bereitgestellt wird.
- Senden Sie Ihre Kommentare per E-Mail an 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.



SC12-3241-01

