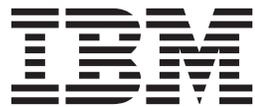


IBM DB2 10.1
for Linux, UNIX and Windows

*Net Search Extender
Verwaltung und Benutzerhandbuch
Aktualisierung: Januar 2013*



IBM DB2 10.1
for Linux, UNIX and Windows

*Net Search Extender
Verwaltung und Benutzerhandbuch
Aktualisierung: Januar 2013*



Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die allgemeinen Informationen in Anhang B, „Bemerkungen“, auf Seite 329 gelesen werden.

Impressum

Diese Veröffentlichung ist eine Übersetzung des Handbuchs
IBM DB2 10.1 for Linux, UNIX, and Windows, Net Search Extender Administration and User's Guide,
IBM Form SC27-3895-01,
herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 1995, 2013

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:
TSC Germany
Kst. 2877
Januar 2013

Inhaltsverzeichnis

Teil 1. Net Search Extender - Übersicht und Konzepte 1

Kapitel 1. Net Search Extender - zentrale Konzepte 3

Skalare SQL-Suchfunktion - Übersicht	6
Suche mit gespeicherten Prozeduren - Übersicht	7
SQL-Tabellenwertfunktion - Übersicht	8
Weitere Konzepte	9
Spaltenumsetzungsfunktion	9
DB2 Net Search Extender-Instanzservices	10
DB2 Net Search Extender-Instanzservices unter Windows	10
Extern gespeicherte Daten	10
Verwaltungstabellen und -sichten	10
Unterstützung für partitionierte Datenbanken	10
Indizes für Kurznamen in föderierten Datenbanken	11
Native XML-Unterstützung	12
Unterstützung für partitionierte Tabellen	12
Teilaktualisierung, die auf Integritätsverarbeitung basiert	13
Unterstützung für separate abgeschirmte Benutzer (Linux und AIX)	14

Kapitel 2. Hauptfunktionen von DB2 Net Search Extender 17

Kapitel 3. Einführung in die db2text-Befehle 19

Teil 2. Installation 21

Kapitel 4. Installation von Net Search Extender in der DB2-Client/Server-Umgebung 23

Kapitel 5. Systemvoraussetzungen für die Installation von Net Search Extender 25

Kapitel 6. Installation von Net Search Extender auf partitionierten DB2-Servern 27

Kapitel 7. Installation von Net Search Extender unter UNIX 29

Kapitel 8. Installation unter Windows 31

Kapitel 9. Verzeichnis- und Dateinamen für Net Search Extender 33

Kapitel 10. Installieren der Outside In-Bibliotheken 35

Kapitel 11. Überprüfen der Installation 37

Überprüfen der Installation unter UNIX	37
Überprüfen der Installation unter Windows	37

Kapitel 12. Deinstallieren von Net Search Extender 39

Deinstallation von Net Search Extender unter UNIX	39
Deinstallation von Net Search Extender unter Windows	39

Teil 3. Konfigurieren von Net Search Extender (NSE) für die Hochverfügbarkeit (HA) 41

Teil 4. Upgrade auf DB2 Net Search Extender 45

Teil 5. DB2 Net Search Extender-Tool für die Indexmigration von 32-Bit auf 64-Bit 49

Teil 6. DB2 Net Search Extender-Tool für die Indexmigration von 32-Bit auf 64-Bit (Windows) 53

Teil 7. Planungsaspekte 57

Kapitel 13. Verzeichnispositionen und Indexspeicher für Net Search Extender . 59

Kapitel 14. Überlegungen zu den Ressourcen für partitionierte DB2-Server . 61

Kapitel 15. Suche mit gespeicherten Prozeduren - Speicherbedarf 63

Net Search Extender-Speicherbedarf für AIX (64-Bit)	63
Speicherbedarf für gespeicherte Prozeduren in Net Search Extender unter Windows (32-Bit und 64-Bit)	64
Net Search Extender-Speicherbedarf für Solaris (64-Bit)	64
Speicherbedarf für gespeicherte Prozeduren in Net Search Extender unter Linux (32-Bit und 64-Bit)	65

Kapitel 16. Aspekte von Tabellen-, Spalten- und Indexnamen	67
Kapitel 17. Dokumentformate und unterstützte Codepages	69
Kapitel 18. Vorbeugende Maßnahmen gegen die Beschädigung von Net Search Extender-Indexdateien.	71
Kapitel 19. Filtersoftware Outside In	73
Kapitel 20. Benutzeraufgabenbereiche	75
Kapitel 21. Erweiterte textverwaltete Zwischenspeicherinfrastruktur für Teilaktualisierungen	77
<hr/>	
Teil 8. Net Search Extender - Verwaltung	79
Kapitel 22. Net Search Extender-Instanzservices	81
Kapitel 23. Starten und Stoppen der Net Search Extender-Instanzservices mithilfe der Befehlszeile	83
Kapitel 24. Net Search Extender-Sperrenservices.	85
Verwenden der Sperrenservices.	85
Anzeigen einer Sperrenmomentaufnahme	87
Kapitel 25. Aktualisierungsservices	89
<hr/>	
Teil 9. Entwicklung: Erstellen und Pflegen eines Textindex	91
Kapitel 26. Aktivieren einer Datenbank	93
Kapitel 27. Inaktivieren einer Datenbank	95
Kapitel 28. Erstellen eines Textindex	97
Erstellen eines Textindex für Binärdatentypen	99
Erstellen eines Textindex für einen nicht unterstützten Datentyp	100
Erstellen eines Textindex für einen Kurznamen mit Indexteilaktualisierung unter Verwendung der DB2-Replikation	101
Erstellen eines Textindex zur Verwendung durch die Suche mit gespeicherten Prozeduren	102
Kapitel 29. Erstellen von Textindizes für Sichten	105

Kapitel 30. Erstellen eines Textindex für bereichspartitionierte Tabellen	107
Kapitel 31. Leistungsaspekte für die Indexierung	109
<hr/>	
Teil 10. Pflegen von Textindizes	111
Kapitel 32. Aktualisieren und Reorganisieren eines Textindex.	113
Kapitel 33. Aktualisieren eines Textindex.	115
Kapitel 34. Ändern eines Textindex	117
Kapitel 35. Löschen von Indexereignissen	119
Kapitel 36. Prüfen eines Textindex (Windows, AIX).	121
Befehl 'checknseindex' (Windows, AIX).	122
Kapitel 37. Löschen eines Textindex	125
Kapitel 38. Anzeigen eines Textindexstatus.	127
Kapitel 39. Backup und Restore von Textindizes	129
Kapitel 40. Entfernen von Dateien aus dem Verzeichnis /tmp	131
<hr/>	
Teil 11. Methoden zum Durchsuchen von Text	133
Kapitel 41. Suchen nach Text mit skalaren SQL-Suchfunktionen.	135
Absetzen einer Abfrage	135
Suchen und Ermitteln der Anzahl gefundener Übereinstimmungen	135
Suchen und Ermitteln der Quote eines gefundenen Textdokuments.	136
Kapitel 42. Angeben von SQL-Suchargumenten	137
Suchen nach Begriffen in beliebiger Reihenfolge	137
Suchen mit den Booleschen Operatoren AND und OR	137
Suchen mit dem Booleschen Operator NOT	138
Suchen nach groben Übereinstimmungen	138
Suchen nach Teilen eines Begriffs (Platzhalterzeichen)	139

Suchen nach Begriffen, die ein Platzhalterzeichen enthalten	139
Suchen nach Begriffen in einer festen Reihenfolge	140
Suchen nach Begriffen im gleichen Satz oder Absatz.	140
Suchen nach Begriffen in Abschnitten strukturierter Dokumente	140
Thesaurussuche	141
Suchen nach numerischen Attributen	141
Freitextsuche	142

Kapitel 43. Zusätzliche Suchsyntaxbeispiele. 143

Kapitel 44. Suchen nach Text mithilfe einer Suche mit gespeicherten Prozeduren. 145

Kapitel 45. Suchen nach Text mit einer SQL-Tabellenwertfunktion 147
Verwenden der Hervorhebungsfunktion 147

Kapitel 46. Suchen in mehreren Spalten 151

Kapitel 47. Verwenden der Textsuche in Outer Joins 153

Kapitel 48. Leistungsaspekte bei der Suche 155

Kapitel 49. Benutzerszenarios 157

Einfaches Beispiel mit der skalaren SQL-Suchfunktion.	157
Einfaches Beispiel mit Verwendung des Cache und der Suche mit gespeicherten Prozeduren	159
Einfaches Beispiel mit der SQL-Tabellenwertfunktion	160

Kapitel 50. Verwenden eines Thesaurus zur Erweiterung von Suchbegriffen 161

Struktur eines Thesaurus	161
Vordefinierte Thesaurusrelationen	162
Definieren eigener Relationen	163
Erstellen und Kompilieren eines Thesaurus	163
Erstellen einer Thesaurusdefinitionsdatei	163
Kompilieren einer Definitionsdatei in ein Thesauruswörterverzeichnis.	165
Thesaurusunterstützung.	165
Vom Thesaurus unterstützte CCSIDs	167
Nachrichten des Thesaurustools	168

Kapitel 51. Net Search Extender-Indexierungskonfiguration. 171

Tokenanalyse	171
Stoppwörter.	172

Sprachen mit Stoppwörterunterstützung	172
Konfiguration	173

Teil 12. Arbeiten mit strukturierten Dokumenten 175

Kapitel 52. Suche in nativ gespeicherten XML-Dokumenten 177

Verwenden des Standarddokumentmodells	177
Verwenden eines angepassten Dokumentmodells	178
XQuery-Unterstützung	179

Kapitel 53. Unterstützung für strukturierte Dokumente. 183

Beschreibung strukturierter Dokumente durch ein Dokumentmodell	183
Beispiel für ein Dokumentmodell.	183
Dokumentmodelle.	184
Textfelder	185
Dokumentattribute	185
Standarddokumentmodelle.	186
Definition eines Dokumentmodells für strukturierte reine Textdokumente	187
Beim Indexieren eines GPP-Dokuments ausgeführte Operationen	188
Definition eines Dokumentmodells für HTML-Dokumente	188
Definition eines Dokumentmodells für XML-Dokumente	191
Definition eines Dokumentmodells für mit Outside In gefilterte Dokumente	196
Beim Indexieren eines Outside In-Dokuments ausgeführte Operationen	197

Kapitel 54. Dokumentmodellreferenz 199

DTD für Dokumentmodelle	199
Semantik von Querverweisausdrücken (XPath)	200
Begrenzungen für Textfelder und Dokumentattribute	202
Attributwerte für Outside In-Tag	203

Teil 13. Referenz 205

Kapitel 55. Verwaltungsbefehle für den Instanzeigner 207

Befehl CONTROL	207
Befehl START	209
Befehl STOP.	211

Kapitel 56. Verwaltungsbefehle für den Datenbankadministrator. 213

Befehl ENABLE DATABASE	213
Befehl DISABLE DATABASE	215
DB2EXTHL, Befehl	217

Kapitel 57. Verwaltungsbefehle für den Texttabelleneigner 219

Befehl ACTIVATE CACHE	219
Befehl ALTER INDEX	221
Befehl CLEAR EVENTS	225
Befehl CREATE INDEX	227
Befehl DEACTIVATE CACHE	242
Befehl DROP INDEX	244
Befehl DB2EXTTH	245
Befehl RESET PENDING	246
Befehl UPDATE INDEX	247
Befehl HELP	251
Befehl COPYRIGHT	252

Kapitel 58. Befehlsreferenz für die Installation und Deinstallation von Net Search Extender unter UNIX 253

Befehl db2nse_install	253
Befehl db2nse_deinstall	254
Befehl db2nsels	254

Kapitel 59. Syntax für Suchargumente 257

Syntax für Suchargumente	257
Suchparameter	260

Kapitel 60. Skalare SQL-Suchfunktion und SQL-Tabellenwertfunktion 267

CONTAINS (Skalarfunktion)	267
NUMBEROFMATCHES (Skalarfunktion)	268
SCORE (Skalarfunktion)	268
Befehl DB2EXT.TEXTSEARCH	269
DB2EXT.HIGHLIGHT	272

Kapitel 61. Funktion zur Suche mit gespeicherten Prozeduren 277

DB2EXT.TEXTSEARCH für die Suche mit gespeicherten Prozeduren	277
--	-----

Kapitel 62. Windows-Systemfehler 281

Kapitel 63. Net Search Extender-Informationskataloge 285

Sichten für Informationen auf Datenbankebene	285
Sichten für Informationen auf Indexebene	288
Sicht 'db2ext.textindexes'	289
Sicht 'db2ext.indexconfiguration'	291
Sicht 'db2ext.textindexformats'	291
Tabellensichten für einen Textindex	292

Ereignissicht	292
Protokolltabellen, Sichten und Kurznamen	293

Kapitel 64. Ursachencodes der Textsuchsteuerkomponente 297

Teil 14. Fehlerbehebung. 305

Kapitel 65. Tracefunktion zur Fehleranalyse 307

Kapitel 66. DB2-Objekte ohne die richtigen Net Search Extender-Befehle löschen. 309

Tabelle löschen	309
Datenbank löschen	309

Kapitel 67. Installationsrückkehrcodes unter Windows. 311

Kapitel 68. Hinweise und Tipps. 313

Teil 15. Anhänge und Schlussteil 317

Anhang A. Übersicht über technische Informationen zu DB2. 319

Bibliothek mit technischen Informationen zu DB2 im Hardcopy- oder PDF-Format	320
Aufrufen der Hilfe für den SQL-Status über den Befehlszeilenprozessor	322
Zugriff auf verschiedene Versionen des DB2 Information Center	322
Aktualisieren des auf Ihrem Computer oder Intranet-Server installierten DB2 Information Center	323
Manuelles Aktualisieren des auf Ihrem Computer oder Intranet-Server installierten DB2 Information Center	324
DB2-Lernprogramme	327
Informationen zur Fehlerbehebung in DB2	327
Bedingungen	328

Anhang B. Bemerkungen 329

Index 333

Teil 1. Net Search Extender - Übersicht und Konzepte

Kapitel 1. Net Search Extender - zentrale Konzepte

Net Search Extender bietet Benutzern und Anwendungsprogrammierern eine Möglichkeit, mithilfe von SQL-Abfragen Volltextdokumente zu durchsuchen, die in DB2-Datenbanken, anderen Datenbanken und Dateisystemen gespeichert sind.

Wichtig: Net Search Extender gilt als veraltet. Die Verwendung des Produkts wird nicht mehr empfohlen; es wird möglicherweise in einem zukünftigen Release entfernt. Verwenden Sie DB2 Text Search als schnelle und vielseitige Methode, um Volltextdokumente, die in DB2-Datenbanken gespeichert sind, mithilfe von SQL- und XQuery-Anweisungen zu durchsuchen. Der Abschnitt zur Migration von Net Search Extender auf DB2 Text Search enthält Details hierzu.

Ein umfassendes Verständnis des Leistungsspektrums von Net Search Extender setzt das Verständnis der Schlüsselbegriffe sowie der verschiedenen verfügbaren Optionen voraus. Darüber hinaus ist ein grundlegendes Verständnis der Konzepte und Begriffe für DB2-Datenbanken erforderlich.

Net Search Extender dient in erster Linie dem Zweck, *Textdokumente* zu durchsuchen, die in der Spalte einer Datenbanktabelle gespeichert sind.

Die Textdokumente müssen eindeutig identifizierbar sein. Net Search Extender verwendet zu diesem Zweck den *Primärschlüssel* der Tabelle.

Die Dokumente können in unterschiedlichen Formaten gespeichert werden. Die Formate umfassen entweder Formate für unstrukturierten Klartext, Formate für strukturierten Text, wie z. B. HTML oder XML, oder proprietäre Dokumentformate, wie z. B. PDF oder Microsoft Office-Dokumentformate. Für Letztere ist zusätzliche Filtersoftware erforderlich, die möglicherweise separat lizenziert werden muss.

Anstelle des sequenziellen Durchsuchens der Textdokumente während der Abfrage, das eine beträchtliche Zeit dauern würde, erstellt Net Search Extender einen *Textindex*, damit Dokumente effizient durchsucht werden können.

Ein Textindex besteht aus wichtigen *Begriffen*, die aus den Textdokumenten extrahiert wurden.

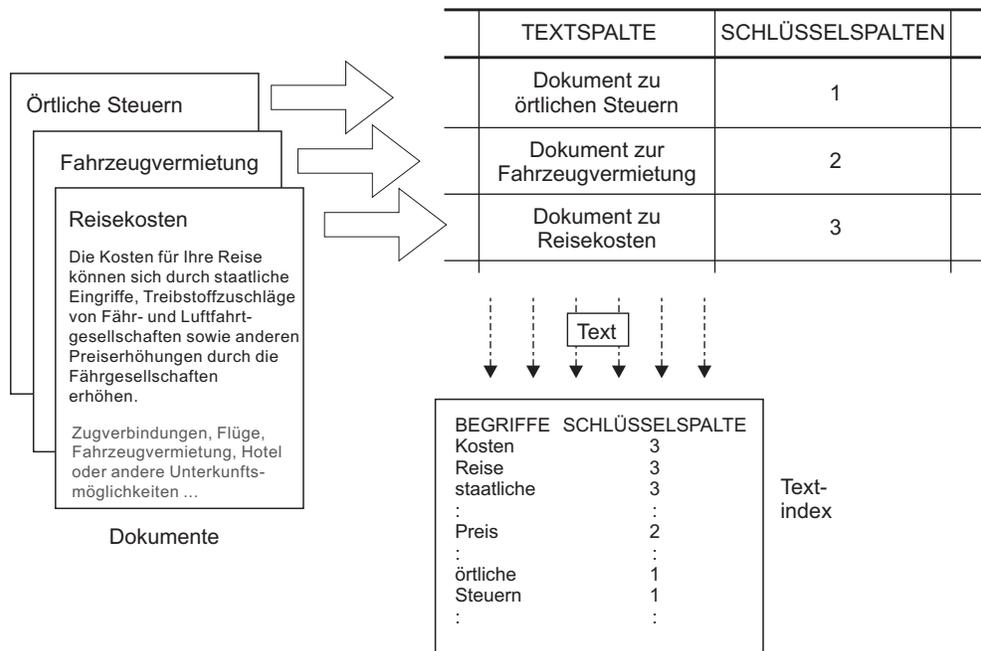


Abbildung 1. Erstellen eines Textindex

Die *Textindexerstellung* ist der Prozess der Definition und Deklaration der Eigenschaften des Index, wie zum Beispiel die Speicherposition des Index. Nach der Erstellung enthält der Textindex bis zu diesem Zeitpunkt keine Daten. Die *Indexaktualisierung* ist der Prozess, durch den Daten über Begriffe und Dokumente zum Textindex hinzugefügt werden. Durch die erste Indexaktualisierung werden dem Index Informationen zu allen Textdokumenten aus der Textspalte hinzugefügt. Die erste Aktualisierung wird als *Anfangsaktualisierung* bezeichnet.

Bei der Verwendung eines Textindex für die Suche sind gewisse Aspekte der Synchronisation zwischen der Tabelle und dem Textindex zu berücksichtigen, da alle Änderungen durch Hinzufügen, Löschen oder Aktualisieren an den Textdokumenten auf den Textindex übertragen werden müssen. Diese Änderungen werden als *Teilaktualisierung* auf den Textindex angewendet.

Net Search Extender unterstützt zwei Optionen, um den Textindex mit der zugehörigen Quellentabelle zu synchronisieren. Die Basissynchronisation in Net Search Extender beruht auf Triggern, die Informationen zu neuen, geänderten und gelöschten Dokumenten automatisch in einer Protokolltabelle speichern. Für jeden Textindex gibt es eine Protokolltabelle.

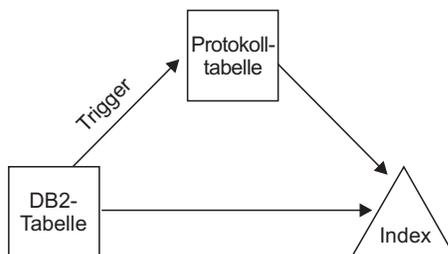


Abbildung 2. Verfahren der Teilaktualisierung mit Triggern

Die Basisoption beruht ausschließlich auf Triggern. Aktualisierungen, die von Triggern nicht erkannt werden, werden daher ignoriert, wie z. B. das Laden von Daten mit dem Befehl **LOAD** oder das Zuordnen von Bereichen zu einer bereichspartitionierten Tabelle bzw. das Aufheben einer solchen Zuordnung. Die erweiterte Synchronisationsoption ermöglicht das Erfassen solcher Änderungen über die Integritätsverarbeitung, indem eine textverwaltete Zwischenspeichertabelle hinzugefügt wird, die Informationen zu neuen und gelöschten Dokumenten speichert, während die Protokolltabelle Informationen zu geänderten Dokumenten speichert, die über einen Trigger erfasst wurden.

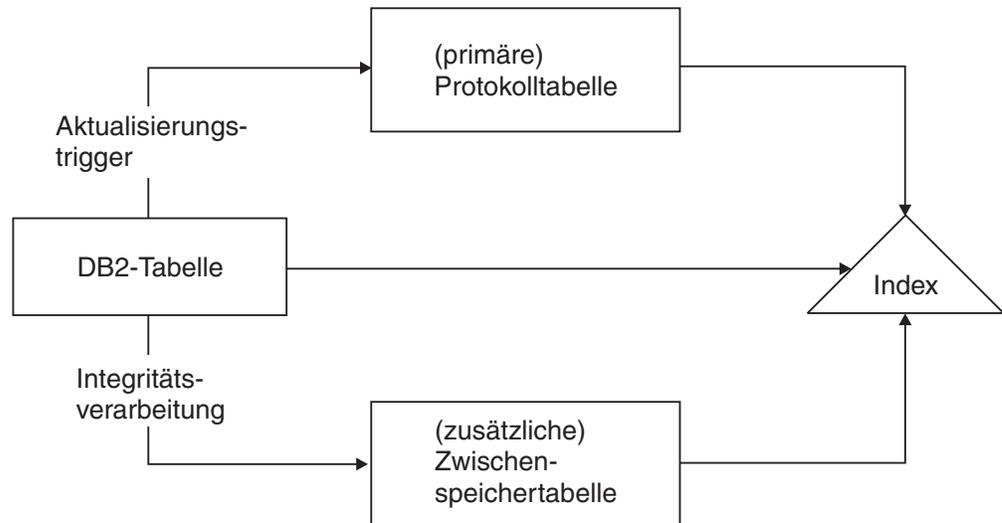


Abbildung 3. Teilaktualisierung mithilfe von Triggern und der Integritätsverarbeitung

Sie können den Textindex mithilfe einer Option für die *manuelle* oder die *automatische* Ausführung aktualisieren. Die Option für die automatische Ausführung arbeitet mit einem Aktualisierungszeitplan, in dem Tage und Uhrzeiten definiert sind.

Beachten Sie, dass keine dieser Optionen den Textindex im Rahmen einer Transaktion synchronisiert, die Textdokumente aktualisiert, löscht und einfügt. Die asynchrone Textindexierung von Net Search Extender verbessert die Leistung und den gemeinsamen Zugriff. Die Aktualisierung wird in einer separaten Transaktion auf eine Kopie eines sehr kleinen Teils des Index angewendet. Der Index wird nur während eines sehr kurzen Zeitraums für den Lesezugriff gesperrt, wenn die Kopie an die Position des Originals gesetzt wird. Dies ist für Suchoperationen nicht sichtbar. Weitere Informationen finden Sie in Kapitel 22, „Net Search Extender-Instanzservices“, auf Seite 81.

Ein Textindex besitzt bestimmte Eigenschaften, wie zum Beispiel die Speicherposition der Indexdatei und die automatischen Aktualisierungseigenschaften. Bei Bedarf können Sie einige dieser Eigenschaften ändern. Dieser Vorgang wird als *Änderung* des Index bezeichnet. Das Ändern des Index beeinflusst keine Indexdaten.

Eine solche Eigenschaft legt z. B. fest, ob der Textindex mit ORDER BY für die Tabellenspalten vorsortiert werden soll. In einem solchen Fall indiziert die Anfangsaktualisierung das Textdokument in der angegebenen Reihenfolge, und die Suchergebnisse werden in dieser Reihenfolge zurückgegeben.

Es können zum Beispiel nach den Buchpreisen vorsortierte Kurzinformationen über Bücher angegeben werden. Wenn Sie nach den preisgünstigsten Büchern über relationale Datenbanksysteme suchen, können Sie Ihre Textsuche so einschränken, dass nur die ersten paar Bücher zurückgegeben werden, da diese in diesem Fall die günstigsten sind. Ohne vorsortierte Indizes müssten Sie nach allen Büchern suchen und diese mit einer Liste der günstigsten Bücher verknüpfen; diese Operation wäre kostenintensiver.

Net Search Extender ermöglicht mehrere vorsortierte Indizes pro Textspalte. Zum Beispiel kann ein Index Bücher nach dem Erscheinungsdatum und ein zweiter Index Bücher nach dem Preis vorsortieren.

In der Regel ist die erste Aktualisierung nach der Erstellung des Textindex eine Anfangsaktualisierung, während es sich bei den nachfolgenden Aktualisierungen um Teilaktualisierungen handelt. Wenn Sie jedoch mit vorsortierten Indizes arbeiten, ist es sinnvoll, die Indexreihenfolge auch bei Aktualisierungen beizubehalten. Dies wird durch die Option RECREATE INDEX ON UPDATE ermöglicht, die bei jeder Durchführung einer Aktualisierung für einen kompletten Rebuild des Index sorgt.

Nachdem der Textindex aktualisiert wurde, können Sie Ihre Textsuchvorgänge mit einer der folgenden Optionen durchführen:

- Eine skalare SQL-Suchfunktion
- Eine Suche mit gespeicherten Prozeduren
- Eine SQL-Tabellenwertfunktion

Da die Suchoptionen verschiedene Ausführungseigenschaften haben, werden sie in den folgenden Abschnitten erläutert.

Skalare SQL-Suchfunktion - Übersicht

Net Search Extender stellt drei skalare Textsuchfunktionen bereit (CONTAINS, NUMBEROFMATCHES und SCORE), die nahtlos in SQL integriert werden.

Aufruf einer skalaren SQL-Suche

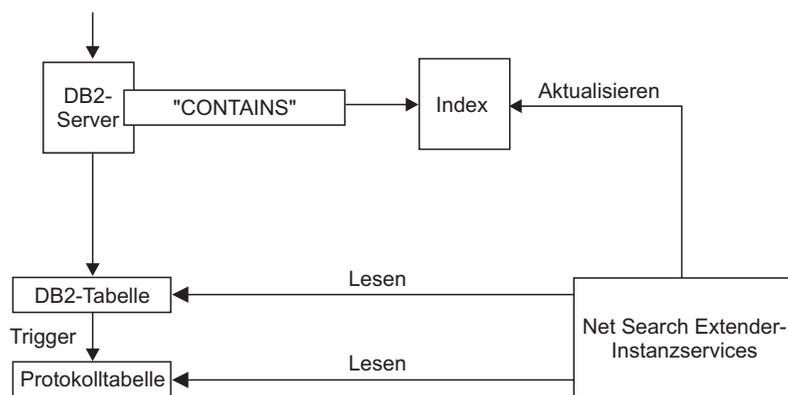


Abbildung 4. Ausführung von Suchoperationen mit einer skalaren SQL-Suchfunktion

Sie können die Suchfunktionen für die gleichen Komponenten verwenden, für die auch SQL-Standardausdrücke in SQL-Abfragen eingesetzt werden würden. Typische Abfragen wären zum Beispiel:

```

SELECT * FROM books WHERE CONTAINS (abstract, "relational databases") = 1
AND PRICE <10

SELECT ISBN, SCORE (abstract, "relational databases") as SCORE
from BOOKS
where NUMBEROFMATCHES (abstract, "relational databases")
>5 AND PRICE <10
order by SCORE

```

Die SQL-Skalarfunktionen im Beispiel geben einen Anzeiger für den Grad der Übereinstimmung der Textdokumente mit einer angegebenen Suchbedingung zurück. Die SELECT-Phase der SQL-Abfrage bestimmt anschließend die Informationen, die an den Benutzer zurückgegeben werden.

Verwenden Sie die SQL-Skalarsuchfunktionen als Standardsuchmethode. Diese Suchfunktionen eignen sich für die Mehrzahl der Situationen, insbesondere wenn der Textsuchausdruck mit anderen Bedingungen kombiniert wird.

Beachten Sie, dass dem DB2-Optimierungsprogramm Schätzwerte für die Anzahl der Textdokumente, für die eine Übereinstimmung mit einem Vergleichselement CONTAINS zu erwarten ist, sowie für den Aufwand für unterschiedliche Zugriffsplanalternativen vorliegen. Das Optimierungsprogramm wählt den günstigsten Zugriffsplan aus.

Suche mit gespeicherten Prozeduren - Übersicht

Die überwiegende Verwendung der Suche mit gespeicherten Prozeduren findet in Hochleistungs- und Hochskalierbarkeitsanwendungen statt, bei denen Abfragen im Mittelpunkt des Interesses stehen, für die keine Textsuchergebnisse mit den Ergebnissen anderer komplexer SQL-Bedingungen verknüpft werden müssen.

In der Regel umfasst die Darstellung von Suchergebnissen beim Endbenutzer einen Aufruf für die Suchfunktion selbst, dem eine Joinoperation für die Benutzertabelle und möglicherweise eine Sortierung der Ergebnisdaten folgt. Diese Operation kann sehr kostenintensiv sein. Es gibt allerdings Situationen, in denen eine Anwendung kostenintensive Plattenoperationen durch die Ausführung der Joinoperationen für vorsortierte Daten vermeiden kann, die im Speicher abgelegt sind. Diese Situationen sind folgende:

- Die dem Benutzer anzuzeigende Datenuntergruppe ist gering.
- Die Datenuntergruppe ist im Vorfeld bekannt.
- Die geplante Sortierreihenfolge ist festgelegt und im Vorfeld bekannt.
- Eine eingestufte Untergruppe der Suchergebnisse ist ausreichend.

Bei der Textindexerstellung müssen Sie angeben, welche Spalten aus der Tabelle oder Sicht an den Endbenutzer zurückgegeben werden sollen. Die Daten werden in einem **Cache** im Hauptspeicher gespeichert. Durch dieses Verfahren können Suchergebnisse der Suche mit gespeicherten Prozeduren äußerst schnell zurückgegeben werden. Der Cache muss **aktiviert** werden, bevor er genutzt werden kann. Dementsprechend ist auch ein Befehl zum **Inaktivieren** vorhanden.

Aufruf der Suche mit der gespeicherten Prozedur TextSearch

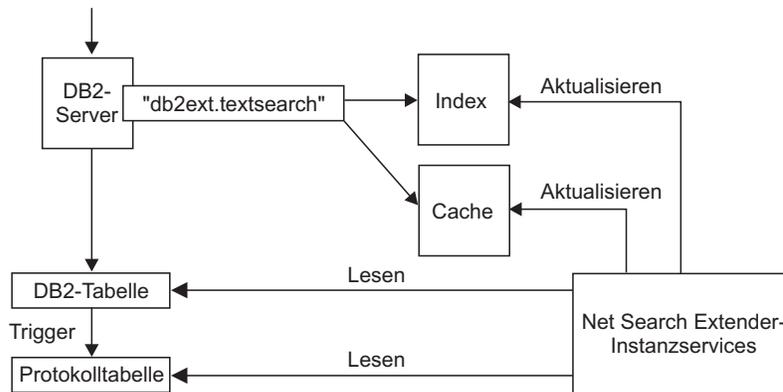


Abbildung 5. Verwenden einer Suche mit gespeicherten Prozeduren

Der Befehl `ACTIVATE` lädt Daten entweder in einen temporären Cache (der bei Aktivierung völlig neu erstellt wird) oder in einen persistenten Cache, der auf der Platte verwaltet wird.

Wenn Sie sich dazu entscheiden, die gespeicherte Prozedur für Suchoperationen zu verwenden, sind sorgfältige Speicherberechnungen erforderlich; es muss beispielsweise berechnet werden, wie viel Speicher benötigt wird und wie viel freier Speicher für Indexaktualisierungen verfügbar sein sollte. Standardwerte finden Sie in Kapitel 15, „Suche mit gespeicherten Prozeduren - Speicherbedarf“, auf Seite 63.

Die gespeicherte Prozedur kann mit Textindizes arbeiten, die für Sichten erstellt werden. Da für Sichten jedoch keine Trigger erstellt werden können, werden Änderungen nicht automatisch erkannt. Sie müssen die geänderten Informationen manuell zur Protokolltabelle hinzufügen oder die Option `RECREATE` verwenden.

Die überwiegende Verwendung der Suche mit gespeicherten Prozeduren findet in Hochleistungs- und Hochskalierbarkeitsanwendungen statt, bei denen Abfragen im Mittelpunkt des Interesses stehen, für die keine Textsuchergebnisse mit den Ergebnissen anderer komplexer SQL-Bedingungen verknüpft werden müssen.

Die funktionellen Hauptunterschiede zu skalaren SQL-Suchfunktionen sind folgende:

- Die Suche mit gespeicherten Prozeduren kann nicht in willkürlichen SQL-Abfragen verwendet werden, sondern stellt eine Abfrage für eine vordefinierte Cachtabelle dar.
- Die Suche mit gespeicherten Prozeduren kann Indizes für Sichten nutzen.
- Die Suche mit gespeicherten Prozeduren kann mehrere vorsortierte Textindizes für eine Spalte nutzen.

SQL-Tabellenwertfunktion - Übersicht

Die SQL-Tabellenwertfunktion stellt einen Kompromiss zwischen den skalaren SQL-Suchfunktionen und der Suche mit gespeicherten Prozeduren dar. Sie können zusammen mit der SQL-Tabellenwertfunktion die Funktion `db2ext.highlight` verwenden, um Informationen darüber zu erhalten, warum ein Dokument als Suchergebnis qualifiziert wurde.

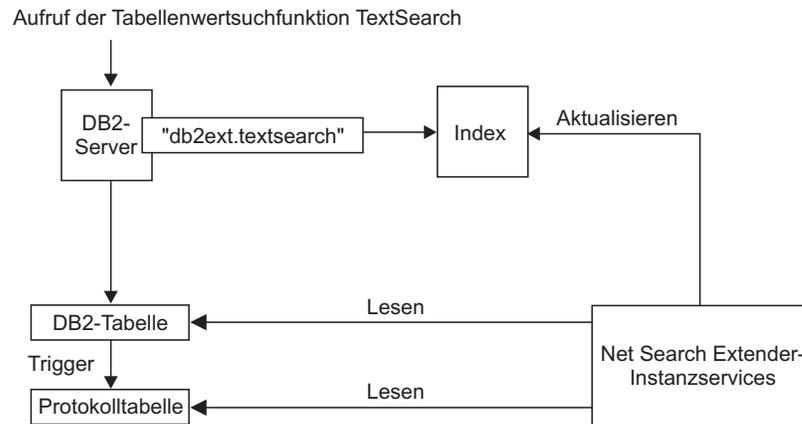


Abbildung 6. Ausführung von Suchoperationen mit einer SQL-Tabellenwertfunktion

Die funktionellen Hauptunterschiede zur Suche mit gespeicherten Prozeduren lauten wie folgt:

- Es ist kein Cache erforderlich (und es wird kein Cache genutzt).
- Die Tabellenwertfunktion kann in willkürlichen SQL-Anweisungen verwendet werden.
- Große Speicherkapazitäten sind zum Vorspeichern von Cachetabelleninhalt nicht erforderlich.

Der funktionelle Hauptunterschied zu den skalaren SQL-Suchfunktionen ist folgender:

- Die SQL-Tabellenwertfunktion kann Indizes für Sichten nutzen.

Verwenden Sie die SQL-Tabellenwertfunktion in den Fällen, in denen Sie normalerweise eine skalare SQL-Funktion verwenden würden, jedoch Textindizes für Sichten verwenden möchten.

Weitere Konzepte

Spaltenumsetzungsfunktion

Sie haben die Möglichkeit, eine eigene Funktion zum Konvertieren eines nicht unterstützten Formats oder Datentyps in ein unterstütztes Format bzw. einen unterstützten Datentyp zu verwenden. Durch Angabe einer benutzerdefinierten Funktion (UDF = User Defined Function) wird als Eingabe das Originaltextdokument abgerufen.

Sie haben die Möglichkeit, eine eigene Funktion zum Konvertieren eines nicht unterstützten Formats oder Datentyps in ein unterstütztes Format bzw. einen unterstützten Datentyp zu verwenden. Durch Angabe einer benutzerdefinierten Funktion (UDF = User Defined Function) wird als Eingabe das Originaltextdokument abgerufen. Die Ausgabe der UDF sollte ein unterstütztes Format aufweisen, das bei der Indexierung verarbeitet werden kann.

Sie können diese Möglichkeit auch zur Indexierung von Dokumenten nutzen, die sich in externen, nicht direkt unterstützten Datenspeichern befinden. In diesem Fall enthält die DB2-Spalte Dokumentverweise, und die Funktion gibt die Dokumentinhalte zurück, die den relevanten Dokumentverweis enthalten.

DB2 Net Search Extender-Instanzservices

Die Net Search Extender-Instanzservices sorgen für die indexspezifischen Sperrservices und die Aktualisierungsservices für Textindizes (sowohl automatische als auch manuelle).

Die Net Search Extender-Instanzservices sorgen für die indexspezifischen Sperrservices und die Aktualisierungsservices für Textindizes (sowohl automatische als auch manuelle).

DB2 Net Search Extender-Instanzservices unter Windows

Wenn eine neue DB2-Instanz erstellt wird, wird der Service DB2EXT für die erste Partition automatisch erstellt.

Später werden DB2EXT-Services hinzugefügt, wenn mit dem Befehl **db2ncrt** neue Partitionen hinzugefügt werden. DB2EXT-Services werden auch hinzugefügt, wenn der Befehl **db2start add dbpartitionnum** ausgeführt wird. In gleicher Weise werden die Befehle **db2ndrop** und **db2nchg** zum Löschen bzw. Ändern von DB2EXT-Services verwendet.

Extern gespeicherte Daten

Extern gespeicherte Textdokumente werden unterstützt, sollten jedoch mit DB2-Kurznamentabellen verwendet werden, um einen Textindex zu erstellen.

In der Mehrheit der Fälle werden die Daten, für die Sie einen Textindex erstellen, in DB2-eigenen Tabellenspalten des Datentyps CLOB oder VARCHAR gespeichert.

Allerdings werden Textdokumente, die extern, zum Beispiel in anderen Datenbanken gespeichert sind, ebenfalls unterstützt. Für Dokumente, die in anderen Datenbanken gespeichert sind, können Sie DB2-Kurznamentabellen verwenden, um einen Textindex zu erstellen.

Sie können die Spaltenumsetzungsfunktion außerdem für Daten verwenden, die in nicht unterstützten externen Datenspeichern abgelegt sind.

Verwaltungstabellen und -sichten

Die Net Search Extender-Tabellen und -Sichten enthalten Informationen zu Textindizes und Eigenschaften.

In Net Search Extender sind mehrere Tabellen und Sichten verfügbar. Diese bieten Informationen zu Textindizes und ihren Eigenschaften.

Unterstützung für partitionierte Datenbanken

Wenn Sie NSE in einer Umgebung mit partitionierten Datenbanken aktivieren und verwalten, müssen Sie eine Reihe von Faktoren berücksichtigen, die sich auf die Datenbank auswirken können.

Beachten Sie beim Aktivieren und Verwalten von NSE in einer Umgebung mit partitionierten Datenbanken die folgenden Faktoren:

- Stellen Sie sicher, dass DB2 vollständig, wie in der DB2-Dokumentation beschrieben, eingerichtet ist. Der NFS-Mount muss mit Rootzugriff und Setuid konfiguriert sein.
- Wenn bei **db2text start** ein Problem auftritt, wird keine detaillierte Nachricht zurückgegeben, die darauf hinweist, welche verfügbaren Partitionen betroffen

sind. Wenn Sie **db2text start** ein zweites Mal absetzen, versucht das System, den Service in den einzelnen Partitionen zu starten. Der Befehl **db2text start** ist erfolgreich, wenn die folgende Nachricht angezeigt wird: CTE0185 Die Aktualisierungs- und Sperrservices sind bereits aktiv.

- Die ID des abgeschirmten Benutzers sollte dieselbe wie die ID des Instanzeigners sein.
- Sie können keine neue Partitionsnummer einfügen oder eine vorhandene Nummer aus `db2nodes.cfg` entfernen, während die NSE-Instanzservices aktiv sind. Dies trifft für alle Befehle zu, die möglicherweise zu Änderungen an `db2nodes.cfg` führen.
- Auf Windows-Plattformen sollte die Datei `db2nodes.cfg` während der Verwendung von NSE in einer Umgebung mit partitionierten Datenbanken für denselben Host nicht sowohl IP-Adressen als auch Hostnamen verwenden.
- Löschen Sie vor dem Ausführen einer Datenumverteilung vorhandene Textindizes, damit unerwartete Ergebnisse und Fehlermeldungen vermieden werden. Wenn die Umverteilungsoperation durchgeführt wird, ohne zunächst die Textindizes in der betreffenden Datenbank zu löschen, können sie auch noch nach Beendigung der Umverteilungsoperation gelöscht werden.

Die Suchfunktionen von Net Search Extender verwenden die Unterstützung für Umgebungen mit partitionierten Datenbanken wie folgt:

- Die Suche mit gespeicherten Prozeduren und die SQL-Tabellenwertfunktion können nur für Tabellen ausgeführt werden, die lokale Tabellen der Koordinatorpartition einer partitionierten Datenbank sind.
- Die skalaren SQL-Suchfunktionen (CONTAINS, NUMBEROFMATCHES und SCORE) können mit wenigen Ausnahmen für Tabellen verwendet werden, die sich über mehrere Partitionen erstrecken.

Sie sollten bei der Durchführung von Suchoperationen in einer Umgebung mit partitionierten Datenbanken Folgendes berücksichtigen:

- Der Ergebnismengengrenzwert (RESULT LIMIT) wird bei der Suche in jeder Partition ausgewertet. Dies bedeutet, dass wenn Sie einen Ergebnismengengrenzwert von 3 angeben und 4 Partitionen verwenden, Sie möglicherweise bis zu 12 Ergebnisse erhalten.
- Der Wert SCORE spiegelt die Relevanz des Dokuments im Vergleich zum Wert SCORE aller Dokumente einer einzigen Partition wider, auch wenn die Abfrage auf mehrere Partitionen zugreift.

Indizes für Kurznamen in föderierten Datenbanken

Sie können einen Textindex für Kurznamen in einer föderierten Datenbank erstellen, der auf Tabellen in einer fernen Datenbank verweist.

In diesem Fall weicht die Funktion der Protokolltabelle (für Indexteilaktualisierungen) von der Rolle dieser Tabelle für einen Index einer regulären Tabelle ab. Anders als bei regulären Tabellen können für Kurznamen keine DB2-Trigger erstellt werden, sodass Änderungsinformationen zu Dokumenten nicht in eine Protokolltabelle eingefügt werden können, die mit Triggern arbeitet. Auch kann die erweiterte textverwaltete Zwischenspeicherinfrastruktur nicht zum Erfassen von Änderungen verwendet werden. Aus diesem Grund gibt es zwei verschiedene Möglichkeiten zur Ausführung von Teilaktualisierungen zum Erstellen eines Index für einen Kurznamen:

- Die Protokolltabelle wird lokal in der föderierten Datenbank erstellt und die Anwendung ist dafür verantwortlich, dass sichergestellt ist, dass die Protokolltabel-

le korrekte Änderungsinformationen zum Kurznamen enthält. Bei DB2-Sichten entspricht dies der Indexteilaktualisierung. Diese Option stellt die Standardeinstellung dar.

- Die DB2-Replikation wurde so konfiguriert, dass Änderungen an der Tabelle, auf die über den zugehörigen Kurznamen verwiesen wird, für ferne DB2-Datenbanken in einer so genannten CD-Tabelle (CD = Change Data) erfasst werden. Bei relationalen Datenbanken anderer Hersteller werden die Änderungen hingegen in einer so genannten CCD-Tabelle (CCD = Consistent Change Data) erfasst. DB2 Net Search Extender kann dann die CD- oder CCD-Tabelle verwenden, anstatt eine Protokolltabelle für einen Index zu einem Kurznamen zu erstellen. In diesem Fall müssen Sie die Merkmale der Erfassungstabelle im Befehl **DB2TEXT CREATE INDEX** angeben.

Native XML-Unterstützung

Durch die vollständige Unterstützung des SQL-XML-Datentyps in Datenbanken mit der Codepage UTF-8 können alle Net Search Extender-Suchfunktionen für XML-Dokumente verwendet werden, die nativ in der Datenbank gespeichert werden. Bitte beachten Sie, dass die Textsuche für SQL-XML-Datentypen in Datenbanken mit anderen Codepages als UTF-8 nicht unterstützt wird und möglicherweise keine Ergebnisse liefert.

Die strukturelle Textsuche in XML-Dokumenten nach Abschnitten (weitere Informationen zum Suchen in **Abschnitten** finden Sie in „Suchparameter“ auf Seite 260) kann durch leistungsfähige XQuery-Verarbeitung der Suchergebnisse erweitert werden. Die Textsuchfunktion von Net Search Extender kann mit der XQuery-Sprachunterstützung des DB2-Datenbankservers zur optimalen Verarbeitung von XML-Dokumenten genutzt werden.

Durch die Volltextsuche in der XQuery-Eingabefunktion `db2-fn:sqlquery()` ist eine Suche in XML-Dokumenten und die Verarbeitung der resultierenden XML-Dokumente mit XQuery möglich:

```
FOR $dept in db2-fn:sqlquery('select Department from MyTable
    where contains(Department, 'sections(/dept/employee/resume) "DB2 XML" '
    = 1')/dept
RETURN $dept/employee/name
```

In diesem Beispiel hat die Spalte "Department" den Datentyp "XML". Weitere Informationen finden Sie in Teil 12, „Arbeiten mit strukturierten Dokumenten“, auf Seite 175.

Es wird ein Beispiel angegeben, in dem Sie sehen können, wie XML-Daten abgefragt werden können. Ziehen Sie `sqllib/samples/extenders/db2ext` zu Rate. Rufen Sie `xmlsample datenbank` auf, um die Datenbank zu füllen, und erstellen und aktualisieren Sie die Indizes. Wenn Sie eine Verbindung zur Datenbank hergestellt haben, können Sie für die Daten mithilfe des Befehls **db2 -tvf xmlsearch** Suchoperationen durchführen.

Unterstützung für partitionierte Tabellen

Sie können einen Textindex für bereichsorientierte Tabellen oder für Tabellen erstellen, die die mehrdimensionale Clusterfunktion in Umgebungen mit einer Datenbankpartition oder in Umgebungen mit mehreren Datenbankpartitionierungen verwenden.

Textindizes werden für alle Kombinationen von Partitionierungsfunktionen unterstützt.

Anmerkung: Bei Version 9.7 wird der Textindex gemäß der Partitionierung der Tabelle über mehrere Datenbankpartitionen hinweg partitioniert. Andere Partitionierungsfunktionen wie Bereichspartitionierung oder mehrdimensionales Clustering haben keinen Einfluss auf die Partitionierung des Textindex.

Teilaktualisierung, die auf Integritätsverarbeitung basiert

Mit der Option **AUXLOG** für den Net Search Extender-Befehl **CREATE INDEX** können Sie steuern, ob für einen Textindex eine zusätzliche Protokollinfrastruktur (eine textverwaltete Zwischenspeichertabelle) verwendet wird.

Diese Zusatztable für die Zwischenspeicherung erfasst Informationen zu neuen und gelöschten Dokumenten über die Integritätsverarbeitung, während Dokumentaktualisierungen über einen Aktualisierungstrigger für die Basistabellenspalte erfasst werden, auf die im Befehl **INDEX CREATE** verwiesen wird.

Für die Verwendung der Option gelten die folgenden Einschränkungen:

- Das Objekt, für das Sie den Textindex erstellt haben, muss eine Basistabelle sein, nicht eine Sicht oder ein Kurzname.
- Sie können die Option **CACHE** nicht angeben.
- Sie können die Option **RECREATE INDEX ON UPDATE** nicht verwenden.
- Sie können keine Replikation zum Steuern von Aktualisierungen verwenden.

Für diese Konfigurationsoption ist bei bereichspartitionierten Tabellen standardmäßig der Wert 'ON' (Ein) festgelegt. Für nicht partitionierte Tabellen ist die Konfigurationsoption auf 'OFF' (Aus) gesetzt.

Das Erfassen von Änderungen für eine Teilaktualisierung des Textindex über die Integritätsverarbeitung erfordert möglicherweise, dass Sie zusätzliche Verwaltungstasks ausführen, nachdem eine Datenbankoperation für die Basistabelle ausgeführt wurde. Sie müssen möglicherweise Nachbearbeitungstasks für den Befehl oder Vorbearbeitungstasks für eine Textindexaktualisierung ausführen, um die Integrität für die Basistabelle oder ihre abhängigen Tabellen festzulegen.

```
db2 "create table test.simple (pk integer not null primary key, comment varchar(48))"
db2 "insert into test.simple values (1, 'blue and red!)"
db2text "create index test.simpleix for text on test.simple(comment)
index configuration(auxlog on) connect to mydb"
db2text "update index test.simpleix for text connect to mydb"
db2 "load from loaddata4.sql of del insert into test.simple"
```

Nach der Ladeoperation ist die Basistabelle blockiert. Beispielsweise führt eine Operation **SELECT** zu folgendem Ergebnis: **SQL0668N** Die Operation ist wegen Ursachencode "1" für Tabelle "TEST.SIMPLE" nicht zulässig. **SQLSTATE=57016**.

Auf die Zwischenspeichertabelle kann zugegriffen werden, aber sie enthält noch keine Informationen zu den geänderten Daten.

```
db2 "set integrity for test.simple immediate checked"
```

gibt Folgendes zurück: **SQL3601W** Die Anweisung bewirkte, dass mindestens eine Tabelle automatisch in den Status 'Festlegen der Integrität anstehend' versetzt wurde. **SQLSTATE=01586**.

Zu diesem Zeitpunkt ist die Zwischenspeichertabelle blockiert und Änderungsoperationen für die Basistabelle werden zurückgewiesen.

```
"insert into test.simple values(15, 'green')"
```

gibt Folgendes zurück: DB21034E Der Befehl wurde als SQL-Anweisung verarbeitet, da es sich um einen gültigen Befehl des Befehlszeilenprozessors handelte. Während der SQL-Verarbeitung wurde Folgendes zurückgegeben: SQL0668N Die Operation ist wegen Ursachencode "1" für Tabelle

```
"SYSIBMTS.SYSTSAUXLOG_IX114555" nicht zulässig. SQLSTATE=57016.
```

```
db2text "reset pending for table test.simple for text connect to mydb"
```

```
db2text "update index test.simpleix for text connect to mydb"
```

Weitere Informationen finden Sie hier: Teil 7, „Planungsaspekte“, auf Seite 57

Unterstützung für separate abgeschirmte Benutzer (Linux und AIX)

Für Net Search Extender ist ein abgeschirmter Benutzeraccount zur Ausführung von benutzerdefinierten Funktionen (UDFs) und gespeicherten Prozeduren außerhalb des vom DB2-Datenbankserver verwendeten Adressbereichs (Speicherbereichs) erforderlich.

Für Net Search Extender wird nun eine separate abgeschirmte Benutzer-ID unterstützt. Diese ID ist nicht mit der ID des DB2-Instanzeigners identisch. Für den Instanzeigner und für den abgeschirmten Benutzer wird eine gemeinsame Gruppe als sekundäre Gruppe hinzugefügt.

Bei der Verwendung der abgeschirmten Benutzer-ID müssen Sie die folgenden Faktoren beachten:

- Wenn der Instanzeigner und der abgeschirmte Benutzer nicht identisch sind, schlagen die Net Search Extender-Verwaltungsbefehle fehl, wenn sie von einem Tabelleneigner oder einem Benutzer mit Steuerungsberechtigung für die Tabelle ausgeführt werden. Wenn beispielsweise der Instanzeigner und der abgeschirmte Benutzer nicht identisch sind und nicht zu einer gemeinsamen sekundären Gruppe gehören, gibt **db2text START** die folgende Fehlermeldung zurück:

```
CTE0312E Für den abgeschirmten Benutzer und den Instanzeigner ist keine gemeinsame sekundäre Gruppe vorhanden.
```

Es kann eine gemeinsame Primärgruppe verwendet werden, aus Sicherheitsgründen wird jedoch eine gemeinsame sekundäre Gruppe empfohlen. Die System-, Root- oder Administratorgruppe sollte nicht als gemeinsame sekundäre Gruppe verwendet werden, da das Sicherheitsrisiko besteht, dass diese Gruppe Eigner von NSE-Dateien sein könnte. Eine neue sekundäre Gruppe sollte zu diesem Zweck erstellt werden.

Anmerkung: NSE-Services können nicht gestartet werden, wenn die gemeinsame sekundäre Gruppe des Instanzeigners und des abgeschirmten Benutzers auf null gesetzt wird.

- Ein abgeschirmter Benutzer benötigt Zugriff auf Index- und Thesaurusdateien.

Anmerkung: Dies wird mithilfe der neuen sekundären Gruppe erreicht. Stellen Sie sicher, dass die 'umask'-Einschränkungen den Gruppenlese- und -schreibzugriff für den abgeschirmten Benutzer zulassen. Als 'umask'-Wert muss 0002 festgelegt werden.

- Die folgenden Verwaltungsbefehle können nur vom Instanzeigner eingegeben werden:
 - CREATE INDEX
 - UPDATE INDEX
 - ALTER INDEX
 - DROP INDEX
 - ACTIVATE CACHE
 - DEACTIVATE CACHE
 - RESET PENDING
 - CLEAR EVENTS
 - DB2EXTTH
 - HELP
 - COPYRIGHT
- Im NSE-Indexverzeichnis befinden sich außer den NSE-Indexdateien keine externen Dateien.
- Unter HPUX muss die Thesaurusdefinitionsdatei erneut kompiliert werden, damit Thesaurussuchvorgänge durchgeführt werden können.
- Die Routinen **NUMBER_DOCS** und **REORG_SUGGESTED** werden in einer Umgebung mit partitionierten Datenbanken nicht unterstützt.

Beispiel

Wenn die ID des Instanzeigners 'db2inst1' lautet und die ID des abgeschirmten Benutzers 'db2fenc1', erstellen Sie eine neue Gruppe mit dem Namen 'video'. Machen Sie diese Gruppe sowohl für den Instanzeigner als auch für den abgeschirmten Benutzer zur sekundären Gruppe.

Dies kann anhand der folgenden Schritte überprüft werden:

```
>id db2inst1
uid=44049(db2inst1) gid=204(search) groups=33(video)
```

```
>id db2fenc1
uid=44048(db2fenc1) gid=100(users) groups=33(video)
```

Kapitel 2. Hauptfunktionen von DB2 Net Search Extender

Zu den Hauptfunktionen von Net Search Extender Version 9.7 gehören Indexierungs- und Suchfunktionen sowie Suchergebnisse.

Net Search Extender Version 9.7 verfügt über folgende Hauptfunktionen:

- Indexieren
 - Schnelle Indexierung sehr großer Datenmengen
 - Dynamische Aktualisierung von Indizes
 - Optional: Speichern von Tabellenspalten im Hauptspeicher während der Indexierung zur Vermeidung kostenaufwendiger physischer Leseoperationen beim Suchvorgang
 - Unterstützung für strukturierte Textformate, wie z. B. HTML und XML
 - Unterstützung der Filtersoftware 'Outside In' eines Drittherstellers
 - Unterstützung für Kurznamentabellen
 - Unterstützung für vorsortierte Textindizes
 - Unterstützung für partitionierte Datenbanken
 - Native XML-Unterstützung
 - Unterstützung für bereichspartitionierte Tabellen und in Gruppen zusammengefasste Tabellen (MDC)
- Suchen
 - Boolesche Operationen
 - Umgebungssuchen nach Wörtern im gleichen Satz oder Absatz
 - Suchen nach "groben Übereinstimmungen" mit Wörtern, die eine ähnliche Schreibweise wie der Suchbegriff haben
 - Suchoperationen mit Platzhalterzeichen für Anfangs-, Mittel- und Endteile von Wörtern, für ganze Wörter und einzelne Zeichen
 - Freitextsuchoperationen nach Dokumenten, die einen bestimmten Text enthalten, wobei das Suchargument in natürlicher Sprache angegeben wird
 - Funktion für Hervorhebung, um zu zeigen, warum ein bestimmtes Dokument als Suchergebnis qualifiziert ist
 - Thesaurus-Unterstützung
 - Einschränken der Suche auf Abschnitte innerhalb von Dokumenten
 - Unterstützung für numerische Attribute
 - Hochgeschwindigkeitssuchvorgänge für eine große Anzahl von Textdokumenten mit zahlreichen gleichzeitig angemeldeten Benutzern
 - Integration in die XQuery-Verarbeitung mit der Funktion `db2-fn:sqlquery()`.
- Suchergebnisse
 - Sie können angeben, wie die Suchergebnisse bei der Indexierung sortiert werden
 - Sie können Untergruppen für Suchergebnisse angeben, wenn große Datenmengen durchsucht und umfangreiche Ergebnislisten erwartet werden
 - Sie können für Suchbegriffe mit einer großen Anzahl von Ergebnissen eine Begrenzung angeben

- Die integrierte SQL-Funktionalität wählt in Verbindung mit dem DB2-Optimierungsprogramm auf der Grundlage der erwarteten Suchergebnisse automatisch den besten Plan aus

Einschränkung: Net Search Extender bietet keine Unterstützung für IBM® DB2 pureScale Feature.

Kapitel 3. Einführung in die db2text-Befehle

Dieser Abschnitt enthält Informationen zu dieser Task, ein Beispiel sowie eine Erläuterung der nächsten Schritte mit den db2text-Befehlen.

Informationen zu diesem Vorgang

Das folgende Beispiel zeigt einen Net Search Extender-Befehl:

```
db2text ENABLE DATABASE FOR TEXT
```

Für jeden Erstellungs- und Verwaltungsbefehl für Indizes können Sie die Datenbank, den Benutzer und das Kennwort angeben.

```
db2text ... connect TO datenbank USER benutzer-id USING kennwort
```

Anmerkung: Beachten Sie, dass die Datenbank durch die Umgebungsvariable **DB2DBDFT** angegeben wird, wenn Sie die Verbindungsoptionen im **db2text**-Befehl nicht angeben.

Zum Anzeigen einer Liste der Befehle geben Sie folgenden Befehl ein:

```
db2text ?
```

Zum Anzeigen der Syntax eines einzelnen Befehls geben Sie folgenden Befehl ein:

```
db2text ?befehl
```

Beispiel

Um beispielsweise die Syntax des Befehls **CREATE INDEX** anzuzeigen, verwenden Sie folgenden Befehl:

```
db2text ?CREATE INDEX
```

db2text gibt 0 zurück, wenn der Befehl erfolgreich verarbeitet wurde. Wenn der Befehl nicht verarbeitet wurde, wird 1 zurückgegeben. Beachten Sie, dass der Befehl **db2text** 0 mit einer Warnung zurückgibt, wenn Dokumentfehler während der Aktualisierung des Index auftreten. Informationen über Dokumentfehler sind in der Ereignistabelle des Index enthalten.

Nächste Schritte

In Abhängigkeit von Ihrem Betriebssystem und Ihrer aktiven Befehlshell interpretiert das System Sonderzeichen, wie z. B. ?, (,), *, ! und ". Wenn der Befehl diese Zeichen enthält, müssen Sie daher Anführungszeichen oder ein Escapezeichen verwenden.

Im folgenden Beispiel eines UNIX-Befehls werden Sonderzeichen verwendet:

```
db2 "SELECT * FROM sample WHERE CONTAINS (DESCRIPTION, '\"enable\"') = 1"
```

Teil 2. Installation

Kapitel 4. Installation von Net Search Extender in der DB2-Client/Server-Umgebung

Die Net Search Extender-Suchfunktion ist in SQL integriert und wird auf dem Server ausgeführt.

Wichtig: Net Search Extender gilt als veraltet. Die Verwendung des Produkts wird nicht mehr empfohlen; es wird möglicherweise in einem zukünftigen Release entfernt. Verwenden Sie DB2 Text Search als schnelle und vielseitige Methode, um Volltextdokumente, die in DB2-Datenbanken gespeichert sind, mithilfe von SQL- und XQuery-Anweisungen zu durchsuchen. Der Abschnitt zur Migration von Net Search Extender auf DB2 Text Search enthält Details hierzu.

Die Net Search Extender-Suchfunktion ist in SQL integriert und wird auf dem Server ausgeführt. Daher müssen Sie Net Search Extender nicht auf dem Client installieren, um Textsuchabfragen absetzen zu können.

Wenn Sie Net Search Extender von einem fernen Client aus verwalten möchten, müssen eine der DB2-Serverversionen sowie Net Search Extender selbst auf der Clientseite installiert sein.

Kapitel 5. Systemvoraussetzungen für die Installation von Net Search Extender

Die Hardware- und Softwaremindestvoraussetzungen für Plattformen für DB2 Version 9.7, die von Net Search Extender unterstützt werden.

Sie müssen DB2 Version 9.7 installieren, bevor Sie Net Search Extender ausführen können.

Net Search Extender wird auf den folgenden Plattformen unterstützt:

- AIX (64-Bit)
- Linux x86 (32-Bit)
- Linux x64 (64-Bit)
- Linux auf zSeries (64-Bit)
- HP-UX auf Itanium-basierten HP Integrity Series-Systemen (64-Bit)
- Solaris UltraSPARC (64-Bit)
- Windows auf x86 (32-Bit)
- Windows on x86 (64-Bit)

Anmerkung: Eine Umgebung mit partitionierten Datenbanken wird von Net Search Extender für (32-Bit)-Plattformen nicht unterstützt.

Der minimale Plattenspeicherplatz für eine typische Net Search Extender-Installation beträgt 50 MB. Abhängig vom Datenvolumen, das indexiert werden soll, gelten möglicherweise zusätzliche Hardwarevoraussetzungen.

Kapitel 6. Installation von Net Search Extender auf partitionierten DB2-Servern

Sie müssen sicherstellen, dass DB2 in jeder Partition korrekt installiert und konfiguriert wurde.

Informationen zu diesem Vorgang

Wichtig: Net Search Extender gilt als veraltet. Die Verwendung des Produkts wird nicht mehr empfohlen; es wird möglicherweise in einem zukünftigen Release entfernt. Verwenden Sie DB2 Text Search als schnelle und vielseitige Methode, um Volltextdokumente, die in DB2-Datenbanken gespeichert sind, mithilfe von SQL- und XQuery-Anweisungen zu durchsuchen. Der Abschnitt zur Migration von Net Search Extender auf DB2 Text Search enthält Details hierzu.

Nach der Installation von DB2 müssen Sie Net Search Extender in jeder Partition installieren.

Anmerkung: Eine abgeschirmte Benutzer-ID, die ungleich der Instanzeigner-ID ist, funktioniert nicht mit partitionierten Datenbanken.

Kapitel 7. Installation von Net Search Extender unter UNIX

Zur Installation von Net Search Extender unter UNIX müssen Sie das Produkt installieren und die DB2-Instanz aktualisieren.

Informationen zu diesem Vorgang

Wichtig: Net Search Extender gilt als veraltet. Die Verwendung des Produkts wird nicht mehr empfohlen; es wird möglicherweise in einem zukünftigen Release entfernt. Verwenden Sie DB2 Text Search als schnelle und vielseitige Methode, um Volltextdokumente, die in DB2-Datenbanken gespeichert sind, mithilfe von SQL- und XQuery-Anweisungen zu durchsuchen. Der Abschnitt zur Migration von Net Search Extender auf DB2 Text Search enthält Details hierzu.

Vorgehensweise

Gehen Sie zur Installation von Net Search Extender unter UNIX wie folgt vor:

1. Installieren Sie die Produktkomponenten.
 - a. Melden Sie sich auf der Zielmaschine als Root an.
 - b. Wechseln Sie in das korrekte Verzeichnis für Ihre Plattform.
 - `cd /cdrom`. Dabei ist *cdrom* der Pfad Ihres CD-ROM-Treibers.
 - `cd plattform`
 - c. Falls eine frühere Version von NSE vorhanden ist, deinstallieren Sie diese.
 - d. Rufen Sie die Datei `./nsetup.sh` auf, und befolgen Sie die Anweisungen in der Anzeige.

Wenn Sie die Lizenzvereinbarung akzeptiert haben, wird eine Liste der möglichen Installationspfade angezeigt. Die auswählbaren Installationspfade sind von der Net Search Extender-Version, die Sie installieren möchten, sowie von den installierten Kopien der DB2-Datenbankprodukte abhängig. Wenn Sie den Pfad für die Installation ausgewählt haben, wird das Produkt in diesem Pfad installiert. In das Verzeichnis `/tmp` wird eine Installationsprotokolldatei mit dem Präfix `db2nsei` geschrieben.
2. Aktualisieren Sie die DB2-Instanz.
 - a. Vergewissern Sie sich, dass Sie als Root arbeiten.
 - b. Verwenden Sie den folgenden Befehl, um von Ihrem Arbeitsverzeichnis in den Pfad zu wechseln, in dem Sie Net Search Extender installiert haben:

```
cd pfad/instance
```

Dabei ist *pfad* der Pfad der DB2-Kopie, in dem Sie Net Search Extender installiert haben.
 - c. Führen Sie `db2iupdt` aus. Verwenden Sie `./db2iupdt db2-instanz`. Hierbei ist *db2-instanz* der Name einer vorhandenen DB2-Instanzbenutzer-ID, über die Sie Net Search Extender verwenden möchten.

Kapitel 8. Installation unter Windows

Bei dieser Methode werden eine Reihe von Befehlszeilenoptionen verwendet. Die Befehlszeilenoptionen, für die ein Parameter angegeben werden muss, müssen ohne Leerzeichen zwischen der Option und dem Parameter angegeben werden.

Informationen zu diesem Vorgang

Wichtig: Net Search Extender gilt als veraltet. Die Verwendung des Produkts wird nicht mehr empfohlen; es wird möglicherweise in einem zukünftigen Release entfernt. Verwenden Sie DB2 Text Search als schnelle und vielseitige Methode, um Volltextdokumente, die in DB2-Datenbanken gespeichert sind, mithilfe von SQL- und XQuery-Anweisungen zu durchsuchen. Der Abschnitt zur Migration von Net Search Extender auf DB2 Text Search enthält Details hierzu.

Bei dieser Methode werden eine Reihe von Befehlszeilenoptionen verwendet. Die Befehlszeilenoptionen, für die ein Parameter angegeben werden muss, müssen ohne Leerzeichen zwischen der Option und dem Parameter angegeben werden.

Vorgehensweise

Zum Installieren unter Windows müssen Sie mit einer Benutzer-ID angemeldet sein, die über Verwaltungsrechte verfügt. Führen Sie anschließend die folgenden Schritte aus:

1. Verwenden Sie das Programm `cdrom:\windows\install\setup.exe`, um die Dateien aus dem Paket auf die Zielmaschine zu übertragen.

Beachten Sie, dass Sie für jede DB2-Instanz eine Benutzer-ID mit Kennwort eingeben müssen, um die korrekten Net Search Extender-Services zu erstellen.

Wenn Sie die Lizenzvereinbarung akzeptiert haben, wird eine Liste der möglichen DB2-Kopienamen angezeigt. Die auswählbaren DB2-Kopienamen sind von der Net Search Extender-Version, die Sie installieren möchten, abhängig. Wenn Sie einen DB2-Kopienamen ausgewählt haben, wird das Produkt in dem Pfad installiert, wo bereits diese DB2-Kopie installiert wurde.

Für eine automatische Installation können Sie die `setup.exe` in zwei unterschiedlichen Modi aufrufen:

RECORDMODE

Es wird eine Antwortdatei zur automatischen Installation erstellt.

Die vordefinierte Antwortdatei `setup.iss` befindet sich im Installationsquellenverzeichnis (wird bei einer unbeaufsichtigten Installation nicht verwendet). Führen Sie den Befehl `setup.exe -r` aus, wenn Sie eine neue Antwortdatei erstellen möchten. Die neue Datei `setup.iss` wird in Ihrem Windows-Verzeichnis erstellt. Kopieren Sie die Datei `setup.iss` in Ihr Installationsquellenverzeichnis. Stellen Sie sicher, dass Sie Ihre alte Antwortdatei gesichert haben.

Beispiel: `setup.exe -r -f1"d:\some_directory\setup.iss"`

SILENTMODE

Automatische Installation.

Stellen Sie sicher, dass sich die Datei `setup.iss` in Ihrem Installationsquellenverzeichnis befindet. Führen Sie den Befehl `setup.exe -s` aus.

Die Installation ist erfolgreich, wenn `ResponseResult` in der Datei `setup.log`, die sich in Ihrem Installationsquellenverzeichnis befindet, auf 0 gesetzt ist.

```
Beispiel: setup.exe -s -f1"d:\some_directory\setup.iss"
-f2"d:\another_directory\mysetup.log"
```

Installationsparameter:

- **/r**: Aufzeichnungsmodus (zeichnet eine Antwortdatei für die unbeaufsichtigte Installation auf)
 - **/s**: Unbeaufsichtigte Installation (führt die Installation automatisch aus)
 - **/f1**: Alternativen Antwortdateinamen angeben (vollständiger Pfad)
 - **/f2**: Alternativen Installationsprotokolldateinamen angeben (vollständiger Pfad)
2. Führen Sie nach der Installation einen Neustart des Systems aus.
 3. Rufen Sie `db2text start` auf, um die DB2 Net Search Extender-Instanzservices zu starten.

Ergebnisse

Jede DB2-Instanz erstellt einen Windows-Dienst. Stellen Sie sicher, dass die DB2-Instanzservices unter einem Benutzerkonto und nicht unter dem Systemkonto ausgeführt werden.

Bei einer partitionierten Instanz wird ein Satz von Windows-Diensten pro Partition erstellt. Die DB2EXT-Instanzservices für die jeweilige Partition müssen unter demselben Benutzerkonto wie die DB2-Instanzservices ausgeführt werden.

Microsoft Cluster Server wird von Net Search Extender nicht unterstützt.

Kapitel 9. Verzeichnis- und Dateinamen für Net Search Extender

Für Net Search Extender-Befehle müssen bestimmte Namen angegeben werden.

Die Verzeichnis- und Dateinamen müssen für alle Net Search Extender-Befehle in SBCS-Zeichen angegeben werden. Die maximale Länge der Pfadnamen (einschließlich Dateiname) beträgt 256 Byte.

Kapitel 10. Installieren der Outside In-Bibliotheken

Wenn Sie Net Search Extender mit der Outside In-Software von Stellent verwenden möchten, müssen Sie die Bibliotheken auf allen Plattformen wie folgt konfigurieren.

Informationen zu diesem Vorgang

Die Outside In-Software von Stellent ist für eine Reihe von Plattformen verfügbar. Einzelheiten hierzu finden Sie in <http://www.oracle.com>.

Vorgehensweise

Wenn Sie Net Search Extender mit der Outside In-Software von Stellent verwenden möchten, müssen Sie die Bibliotheken auf allen Plattformen wie folgt konfigurieren:

- Unter Windows: Stellen Sie sicher, dass das Verzeichnis, in dem sich die Bibliotheken befinden, zur Umgebungsvariablen PATH hinzugefügt wird.
- Unter UNIX: Fügen Sie die Outside In-Bibliotheken zum Installationsverzeichnis der DB2-Bibliotheken (`lib`) hinzu.

Kapitel 11. Überprüfen der Installation

Überprüfen der Installation unter UNIX

Es muss überprüft werden, ob Net Search Extender korrekt installiert ist.

Vorgehensweise

Führen Sie die folgenden Schritte aus, um zu überprüfen, ob Net Search Extender korrekt installiert ist.

1. Führen Sie diese Arbeitsschritte aus, um das Verwaltungsscript **nsesample** zum Einrichten der Textindizes aufzurufen:
 - a. Wechseln Sie zu `ausgangsverzeichnis_des_instanzeigners/sql1lib/samples/extenders/db2ext`.
 - b. Rufen Sie `./nsesample ihredb` auf. Beachten Sie hierbei, dass dieser Befehl die Datenbank erstellt, wenn diese nicht bereits vorhanden ist.
 - c. Prüfen Sie die generierte Ausgabedatei `nsesample.log` in Ihrem Ausgangsverzeichnis.
2. Rufen Sie einige Beispielabfragen zur Ausführung in demselben DB2-Befehlsfenster auf:
 - a. Stellen Sie mithilfe des Befehls `db2 connect to ihredb` eine Verbindung zu Ihrer Datenbank her.
 - b. Führen Sie die Beispielabfragen mithilfe des Befehls `db2 -tvf search` aus.
 - c. Prüfen Sie die Ergebnisse der Abfragen im Script. Beachten Sie dabei, dass jede Abfrage einen oder zwei Treffer zurückgeben sollte.

Ergebnisse

Wenn die Protokolldatei `nsesample_partitioned.log` keine Fehler enthält und alle Abfragen funktionieren, war die Installation von Net Search Extender erfolgreich.

Verwenden Sie für eine partitionierte Datenbank das folgende Überprüfungsbeispiel:

```
nsesample_name_der_partitionierten_datenbank [knotennummer][dateiname_des_tabellenbereichs]
```

Überprüfen der Installation unter Windows

Es muss überprüft werden, ob Net Search Extender korrekt installiert ist.

Vorgehensweise

Führen Sie die folgenden Schritte aus, um zu überprüfen, ob Net Search Extender korrekt installiert ist.

- Umgebung ohne partitionierte Datenbanken:
 1. Richten Sie die Beispieltextrindizes wie folgt ein:
 - a. Rufen Sie **db2cmd** auf, um ein DB2-Befehlsfenster zu öffnen.
 - b. Wechseln Sie zu `sql1lib\samples\extenders\db2ext`.
 - c. Rufen Sie vom DB2-Befehlsfenster aus `nsesample.bat ihredb` auf. Beachten Sie hierbei, dass dieser Befehl die Datenbank erstellt, wenn diese nicht bereits vorhanden ist.

- d. Prüfen Sie die generierte Ausgabedatei `nseample.log` im aktuellen Verzeichnis.
2. Rufen Sie die folgenden Beispielabfragen zur Ausführung im DB2-Befehlsfenster auf:
 - a. Stellen Sie eine Verbindung zu Ihrer Datenbank mithilfe des Befehls `db2 connect to ihredb` her.
 - b. Führen Sie die Beispielabfragen mithilfe des Befehls `db2 -tvf search` aus.
 - c. Prüfen Sie die Ergebnisse der Abfragen im Script. Beachten Sie dabei, dass jede Abfrage einen oder zwei Treffer zurückgeben sollte.

Wenn die Protokolldatei `nseample.log` keine Fehler enthält und alle Abfragen funktionieren, war die Installation von Net Search Extender erfolgreich.

- Umgebung mit partitionierten Datenbanken: Richten Sie die Beispieltextindizes folgendermaßen ein.
 1. Rufen Sie `db2cmd` auf, um ein DB2-Befehlsfenster zu öffnen.
 2. Wechseln Sie zu `<sql1lib>\samples\extenders\db2ext`.
 3. Rufen Sie vom DB2-Befehlsfenster aus `nseample_partitioned.bat <ihredb>` auf.
 4. Prüfen Sie die generierte Ausgabedatei `nseample_partitioned.log` im aktuellen Verzeichnis.

Wenn die Protokolldatei `nseample_partitioned.log` keine Fehler enthält und alle Abfragen funktionieren, war die Installation von Net Search Extender erfolgreich.

Kapitel 12. Deinstallieren von Net Search Extender

Wenn Sie Net Search Extender sowie alle Net Search Extender-Indizes dauerhaft von Ihrem System entfernen möchten, müssen Sie zunächst alle Datenbanken mit Net Search Extender-Indizes inaktivieren und anschließend lediglich Net Search Extender entfernen.

Deinstallation von Net Search Extender unter UNIX

Zur Deinstallation von Net Search Extender auf UNIX-Betriebssystemen müssen Sie die Datenbank inaktivieren, die Instanzen stoppen und den Befehl `db2nse_deinstall` eingeben.

Vorgehensweise

Führen Sie die folgenden Schritte aus, um Net Search Extender auf UNIX-Betriebssystemen ordnungsgemäß zu deinstallieren:

1. Für die einzelnen DB2-Instanzen, von denen Sie Net Search Extender entfernen möchten:
 - a. Wechseln Sie zur Benutzer-ID der DB2-Instanz.
 - b. Wenn Sie Net Search Extender auf dieser Instanz nicht länger verwenden wollen, sollten Sie die Indizes löschen und die Datenbank inaktivieren, bevor Sie Net Search Extender deinstallieren:

```
db2text disable database for text connect to datenbankname
```
 - c. Stoppen Sie die DB2 Net Search Extender-Instanz.
 - d. Stoppen Sie die DB2-Instanz.
2. Vergewissern Sie sich, dass Sie als Root arbeiten.
3. Wechseln Sie von Ihrem Arbeitsverzeichnis in den DB2-Pfad, aus dem Sie Net Search Extender entfernen möchten. Beispiel: `cd /opt/IBM/db2/V10.1/install`.
4. Setzen Sie den Befehl `./db2nse_deinstall` ab. Details zur Syntax des Befehls finden Sie in „Befehl `db2nse_deinstall`“ auf Seite 254.

Deinstallation von Net Search Extender unter Windows

Zum Deinstallieren von Net Search Extender unter Windows-Betriebssystemen müssen Sie die Datenbanken inaktivieren, die Instanzen stoppen und das Programm entfernen.

Vorgehensweise

Führen Sie die folgenden Schritte aus, um Net Search Extender ordnungsgemäß unter Windows zu deinstallieren:

1. Führen Sie für jede Datenbank den Befehl `db2text disable database for text connect to datenbankname` aus.
2. Stoppen Sie die DB2-Instanz.
3. Wählen Sie **Einstellungen** > **Systemsteuerung** > **Software** aus. Wählen Sie in der Liste den Net Search Extender-Eintrag `KOPIENAME` aus, der mit dem DB2-`KOPIENAMEN` übereinstimmt, der Net Search Extender während der Installation zugeordnet wurde.
4. Klicken Sie auf **Entfernen**.

Teil 3. Konfigurieren von Net Search Extender (NSE) für die Hochverfügbarkeit (HA)

DB2 Net Search Extender kann mithilfe der gemeinsamen Verwendung von Indizes durch die Hochverfügbarkeitsknoten sowie Net Search Extender-Indexbackup- und -restoreoperationen für die Unterstützung der Hochverfügbarkeit konfiguriert werden. Net Search Extender-Volltextindizes bestehen aus Daten, die in einer DB2-Datenbank sowie in einigen externen Dateien im Dateisystem gespeichert sind. Nur Net Search Extender-Daten innerhalb der Datenbank werden bei einer Funktionsübernahme wiederhergestellt, wenn DB2 für die Hochverfügbarkeit konfiguriert ist. Die externen NSE-spezifischen Dateien müssen mit dem Funktionsübernahmeknoten gemeinsam verwendet werden, wobei eine dem jeweiligen Benutzerszenario und der jeweiligen Plattform entsprechende Technologie zur gemeinsamen Dateinutzung anzuwenden ist. Die externen Dateien werden bei einer Unterbrechung von Indexaktualisierungsoperationen nicht wiederhergestellt, was zu einer Beschädigung der Indexdateien führt. Für die Dateien muss ein Backup durchgeführt werden, damit sie manuell wiederhergestellt werden können.

Die Unterbrechung einer Indexaktualisierung kann zu einer nicht behebbaren und nicht vorhersehbaren Beschädigung des Index führen. Wie schwerwiegend die Beschädigung ist, hängt von den betroffenen Indexdateien und der Indexoperationsphase zum Zeitpunkt der Unterbrechung ab. Für einige der Indexdateien werden die Aktualisierungen auch direkt vorgenommen, nicht für die entsprechende Kopie, sodass ein Rollback schwieriger ist. Wenn also eine Funktionsübernahme während einer Indexaktualisierung stattfindet, müssen beschädigte Indexdateien von der letzten erfolgreichen Indexaktualisierungsoperation wiederhergestellt werden; diese werden als Momentaufnahmen des Indexverzeichnisses gespeichert.

Hochverfügbarkeitskonfigurationen verhindern, dass Indexdateien, die sich im gemeinsam genutzten Speicher befinden, in einen inkonsistenten Status versetzt werden, falls eine Indexaktualisierung während einer Funktionsübernahme unterbrochen wird. Datenbankobjekte auf dem Übernahmesystem können dazu verwendet werden, die Indexdateien wieder in einen konsistenten Status zu versetzen.

Falls die Dateimomentaufnahme von einer Plattform nicht unterstützt wird, muss eine entsprechende Technologie zur gemeinsamen Dateisystemnutzung, die für die betreffende Plattform anwendbar ist, für gemeinsame NSE-Indexordner bzw. -laufwerke verwendet werden.

Momentaufnahmen des Indexverzeichnisses

1. Alle Net Search Extender-Indexdateien müssen in dedizierten Dateisystemen gespeichert sein, damit ein Backup und Restore der neuesten Indexdateien möglich ist. In diesem Dateisystem dürfen keine anderen Daten gespeichert werden.
2. Jeder Index muss sich in einem eigenen Dateisystem befinden. Alternativ dazu können Indizes Dateisysteme auch gemeinsam nutzen; die Aktualisierungsplanung für solche Indizes wird jedoch so serialisiert, dass nie zwei Aktualisierungen gleichzeitig stattfinden. Die Anzahl unterschiedlicher Dateisysteme für Net Search Extender-Indizes muss dann der Anzahl paralleler Aktualisierungsprozesse angepasst werden, die das System ausführen kann.
3. Die Menge des für eine Momentaufnahme erforderlichen Speicherplatzes ist zunächst sehr gering, wächst jedoch in der Regel mit der Veränderung des Datei-

systeminhalts. Vergewissern Sie sich, dass genügend Dateispeicherplatz für die Momentaufnahme im Indexdateisystem vorhanden ist. Überwachen Sie die Dateispeicherbelegung, um sicherzustellen, dass der Speicherplatz für die Momentaufnahmen ausreicht.

Vorbereiten der Funktionsübernahme

Indizes befinden sich in dem Speicherbereich, der von den Hochverfügbarkeitsknoten gemeinsam genutzt wird. Auf jede Indexaktualisierung und jede geplante Aktualisierung muss unmittelbar eine Momentaufnahme des zugehörigen Indexverzeichnisses folgen. Diese Anweisungen können in ein Script eingebunden und von einem externen Scheduler ausgeführt werden, wie in den folgenden Schritten beschrieben:

1. Überprüfen Sie, ob sich die Indexdateien in dem von den Hochverfügbarkeitsknoten gemeinsam genutzten Bereich befinden.
2. Überprüfen Sie den DB2 Net Search Extender-Status über die Tabelle `db2ext.tcommandlock` und das Verzeichnis `work`.
3. Führen Sie die Momentaufnahmenprozedur durch, um eine Momentaufnahme des Net Search Extender-Indexdateisystems im gemeinsam genutzten Speicher zu erstellen.
4. Rufen Sie den Net Search Extender-Befehl **UPDATE INDEX** auf.
5. Entfernen Sie die selbstdefinierte Markierung nach Abschluss der Indexaktualisierung.

Anmerkung: Da die native geplante Indexaktualisierung von Net Search Extender nur den Befehl **DB2TEXT UPDATE INDEX** aufrufen kann, inaktivieren Sie sie, indem Sie für **UPDATE FREQUENCY** den Wert **NONE** definieren. Verwenden Sie stattdessen die betriebssystemspezifische Indexaktualisierungsplanung, wie z. B. den Befehl **CRON** unter UNIX und Linux bzw. den Befehl **AT** auf Windows-Betriebssystemen. Mit diesen Befehlen wird das Wrapper-Script in den angegebenen Intervallen aufgerufen, wobei ein **crontab**-Eintrag für jeden Index vorhanden ist, für den ein automatisierter Aktualisierungsplan existiert. Auf diese Weise wird sichergestellt, dass für alle Indizes im Dateisystem aktuelle Momentaufnahmen von der neuesten erfolgreichen Aktualisierung im gemeinsamen Speicher vorhanden sind.

Indexmerkmale während der Funktionsübernahme

Entscheidend für die Indexrecovery ist, festzustellen, ob der Index bei der Funktionsübernahme beschädigt wurde. Dies erfordert einen Rückgriff auf den letzten bekannten fehlerfreien Status dieses Index und kann mithilfe des folgenden Net Search Extender-Indexaktualisierungsprozesses festgestellt werden:

- Jedes Indexaktualisierung wird intern in ein Einfüge- und Löschoptionspaar in der Tabelle `db2ext.tcommandlocks` eingebunden.
- Um gleichzeitige Verwaltungsbefehle für diesen Index zu verhindern, wird zu Beginn der Indexaktualisierung eine Zeile mit dem benannten Index, einer Zeitmarke und der Art der Operation in dieser Tabelle erstellt. Vor der Beendigung der Aktualisierung wird die Zeile wieder aus der Tabelle entfernt, sodass der Index für weitere Verwaltungsbefehle zur Verfügung steht.
- Wenn während einer Funktionsübernahme keine Indexaktualisierung stattfindet, enthält die Tabelle `db2ext.tcommandlocks` keine Zeilen und es ist keine weitere Aktion erforderlich. Alle in der Protokolltabelle gespeicherten Daten stehen durch die Hochverfügbarkeitsunterstützung sofort auf dem Übernahmesystem zur Verfügung und sind für die nächste reguläre Indexaktualisierung bereit.

- Wenn eine Funktionsübernahme während einer Indexaktualisierung stattfindet, enthält die Tabelle `db2ext.tcommandlocks` auf dem Übernahmeknoten eine Zeile für jeden Index, der zum Zeitpunkt der Funktionsübernahme an einer Aktualisierung beteiligt war. Es können mehrere Indizes betroffen sein; diese entsprechen jeweils einer Zeile in der Tabelle `db2ext.tcommandlocks`, sodass jede Operation für jede Zeile wiederholt werden muss. Für den Restore der Momentaufnahme muss dann eine manuelle Recovery initiiert werden. Jeder betroffene Index wird durch den Sperreintrag der Tabelle vor weiteren (geplanten oder manuellen) Aktualisierungen geschützt.
- Überprüfen Sie, ob die Einträge in der Protokolltabelle noch vorhanden sind. Vergleichen Sie die Zeitmarke des ältesten Protokolltabelleneintrags für den Index mit dem neuesten CTE0003-Eintrag in der Ereignistabelle des Index. Wenn der älteste Protokolltabelleneintrag neuer ist als der neueste CTE0003-Eintrag, wurde bereits vor der Funktionsübernahme eine Bereinigung der Protokolltabelle durchgeführt, doch der Eintrag in `db2ext.tcommandlocks` konnte noch nicht gelöscht werden. In diesem Fall ist der Index unbeschädigt. Führen Sie daher keinen Restore der Momentaufnahme durch, sondern entfernen Sie lediglich den Eintrag in `db2ext.tcommandlocks` und setzen Sie die normale Verarbeitung fort.
Wenn der älteste Protokolltabelleneintrag älter ist als der neueste CTE0003-Eintrag, muss der Index mithilfe der Momentaufnahme wiederhergestellt werden.

Restore des Index mithilfe der Momentaufnahme

1. Entfernen Sie alle Indexdateien im Indexverzeichnis des betroffenen Index. Beachten Sie, dass während dieses Zeitraums alle Suchvorgänge für diesen Index fehlschlagen, und stoppen Sie Net Search Extender.

```
rm -rf /myWORK/NODE0000/TMP_IX300608/*
```
2. Ersetzen Sie das leere Verzeichnis mit dem Inhalt der Momentaufnahme. Dies nimmt einige Zeit in Anspruch, da ein physischer Kopiervorgang der Dateien erforderlich ist.

```
rm -rf /myINDEX
mount -o snapshot /dev/fs1v06 /mnt/
cp -pR /mnt/* /myINDEX
```
3. Nach dem Restore des Indexverzeichnisinhalts entfernen Sie die Zeile, die dem Index entspricht, manuell aus der Tabelle `db2ext.tcommandlocks`.

```
db2 "delete from db2ext.tcommandlocks"
```
4. Wiederholen Sie den oben beschriebenen Vorgang für alle betroffenen Indizes.
5. Starten Sie nach dem Abschluss dieser Schritte Net Search Extender erneut. Der reguläre Betrieb kann nun auf dem Übernahmeknoten fortgesetzt werden.

Der Net Search Extender-Inhalt der Protokolltabelle bleibt intakt und ein neuer Aufruf von **DB2TEXT INDEX UPDATE** verarbeitet ihn wie zuvor. Einige manuelle Bereinigungsaktionen müssen möglicherweise in der Ereignistabelle durchgeführt werden, da diese Einträge enthalten kann, die während der ursprünglichen Indexaktualisierungsoperation erstellt wurden.

Teil 4. Upgrade auf DB2 Net Search Extender

Vor einem Upgrade von Net Search Extender auf DB2 Net Search Extender muss ein Upgrade für den DB2-Server und die DB2-Instanz (unter Linux und UNIX) einschließlich aller Datenbanken auf DB2 Version 10.1 erfolgreich durchgeführt werden.

Vorbereitende Schritte

Wichtig: Net Search Extender gilt als veraltet. Die Verwendung des Produkts wird nicht mehr empfohlen; es wird möglicherweise in einem zukünftigen Release entfernt. Verwenden Sie DB2 Text Search als schnelle und vielseitige Methode, um Volltextdokumente, die in DB2-Datenbanken gespeichert sind, mithilfe von SQL- und XQuery-Anweisungen zu durchsuchen. Der Abschnitt zur Migration von Net Search Extender auf DB2 Text Search enthält Details hierzu.

- Sie müssen vor dem Upgrade für alle Textindexverzeichnisse und Indexunterverzeichnisse ein Backup durchführen. Entsprechende Informationen finden Sie in Kapitel 39, „Backup und Restore von Textindizes“, auf Seite 129.

Informationen zu diesem Vorgang

Das Upgrade auf Net Search Extender wird ab Net Search Extender Version 9.5 bzw. Version 9.7 unterstützt.

Vorgehensweise

Gehen Sie wie folgt vor, um ein Upgrade auf DB2 Net Search Extender durchzuführen:

1. Führen Sie für Ihren DB2-Server, auf dem Net Search Extender installiert ist, ein Upgrade durch. Verwenden Sie dazu eine der folgenden Tasks:
 - Upgrade für DB2-Server durchführen (Windows) in *Upgrade auf DB2 Version 10.1*
 - Upgrade für DB2-32-Bit-Server auf 64-Bit-Systeme durchführen (Windows) in *Upgrade auf DB2 Version 10.1*
 - Upgrade für DB2-Server durchführen (Linux and UNIX) in *Upgrade auf DB2 Version 10.1*

Das Durchführen des Upgrades für Ihre Datenbank ist Teil der entsprechenden Tasks. Wenn Sie über externe, nicht abgeschirmte Routinen unter Linux oder UNIX verfügen, die nicht von den DB2-Enginebibliotheken abhängig sind, definiert der Befehl **UPGRADE DATABASE** Ihre externen Routinen neu als abgeschirmt und nicht threadsicher. Der Abschnitt zum Durchführen eines Upgrades für C-, C- und COBOL-Routinen in der Veröffentlichung Upgrade auf DB2 Version 10.1 enthält Einzelheiten zur sicheren Ausführung Ihrer Routinen im neuen Multithreaddatenbankmanager. Die Net Search Extender-Funktionen mit dem Schemanamen DB2EXT, die während des Upgrades für die Datenbank geändert wurden, werden in Schritt 6 des Scripts db2extmdb als NOT FENCED und THREADSAFE neu definiert.

2. Installieren Sie DB2 Net Search Extender.

Im Gegensatz zu DB2-Datenbanken unterstützt DB2 Net Search Extender keine Upgradeinstallation.

Wenn die installierte DB2-Kopie mithilfe der DB2-Option 'Upgradeinstallation' auf Version 10.1 umgestellt wurde, befindet sich die installierte Kopie von DB2 Net Search Extender noch auf dem Stand der Vorversion.

Wenn Sie versuchen, DB2 Net Search Extender auf eine bereits vorhandene frühere Version von DB2 Net Search Extender zu installieren, erhalten Sie eine Fehlermeldung, in der Sie informiert werden, dass die vorhandene DB2 Net Search Extender-Installation zuerst entfernt werden muss. Deinstallieren Sie in diesem Fall die frühere Version von DB2 Net Search Extender, bevor Sie DB2 Net Search Extender installieren. Unter Windows-Betriebssystemen müssen Sie einen Warmstart des Systems durchführen, nachdem Sie DB2 Net Search Extender deinstalliert haben.

3. Ein Instanzupgrade für DB2 Net Search Extender ist nur unter Linux und UNIX möglich. Unter Windows-Betriebssystemen muss dieser Schritt ignoriert werden. Melden Sie sich für das Upgrade der Instanz als Root an und führen Sie das Script **db2extimigr** mit der folgenden Syntax aus:

```
db2-verzeichnis/instance/db2extimigr [-h|-?] InstanceName
```

Hierbei ist *db2-verzeichnis* das Verzeichnis, in dem Ihre Kopie von DB2 Version 10.1 installiert ist.

4. Prüfen Sie unter Linux und UNIX nach einem erfolgreichen DB2 Net Search Extender-Instanzupgrade die Installation, bevor Sie mit dem Datenbankupgrade fortfahren. Unter Windows-Betriebssystemen können Sie die Installation sofort prüfen.

Anmerkung: Wenden Sie die Beispielskripts von DB2 Net Search Extender nicht für eine Datenbank an, für die kein Upgrade auf DB2 Net Search Extender durchgeführt wurde. Eine sichere Methode ist die Erstellung einer Datenbank für die Überprüfung. Weitere Details finden Sie in „Überprüfen der Installation unter UNIX“ auf Seite 37 und „Überprüfen der Installation unter Windows“ auf Seite 37.

5. Führen Sie für jede Datenbank, die in einem früheren Release für Net Search Extender aktiviert war, ein Upgrade durch. Die Schritte a) und c) sind momentan erforderlich, um ein bekanntes Problem bei **db2extmdb** zu beheben (doppelte Einträge in der Sicht DB2EXT.DBDEFAULTS nach dem Datenbankupgrade). Führen Sie die Datenbankupgradeschritte wie folgt aus:

- a. Melden Sie sich beim DB2-Server als Instanzeigner an.

Sie müssen in der Lage sein, die Instanzservices von Net Search Extender erfolgreich zu starten und zu stoppen; Sie benötigen die Berechtigung DBADM mit DATAACCESS für die Datenbank, für die das Upgrade durchgeführt werden soll. Unter Windows-Betriebssystemen muss der Instanzbenutzer Mitglied der Gruppe lokaler Administratoren sein.

Unter Windows-Betriebssystemen müssen Sie die Verarbeitung von einem DB2-Befehlsfenster aus vornehmen, das mit vollständigen Administratorberechtigungen ausgeführt wird. Weitere Details hierzu finden Sie im Abschnitt Benutzerzugriffssteuerungsfunktion.

Unter den Betriebssystemen Windows 2008 und Windows Vista (und nachfolgenden Betriebssystemen) kann der Befehl **db2extmdb** nicht ausgeführt werden, wenn er von einem Befehlsfenster aus gestartet wird, das mit Standardbenutzerrechten ausgeführt wird. ('CTE0228 Der Benutzer hat keine ausreichenden Zugriffsberechtigungen auf der Betriebssystemebene.')

- b. Führen Sie das Script **db2extmdb** aus, um das Upgrade für die Datenbank durchzuführen, die Sie für Net Search Extender aktiviert haben. Die Syntax lautet wie folgt:

db2extmdb *datenbankname*

Vermeiden Sie es, während der Ausführung dieses Scripts Änderungen an Benutzertabellen mit Textindizes vorzunehmen. Sie können den Befehl für jede Datenbank wiederholen, die über Net Search Extender-Indizes verfügt.

Alle Upgradeschritte werden in der Datei *db2extm^{datenbankname}.log* protokolliert, die sich in einem der folgenden Verzeichnisse befindet:

- *instanzausgangsverzeichnis/sql1lib/db2ext/* unter Linux- und UNIX-Betriebssystemen
- **db2-pfad**\db2ext\ unter Windows-Betriebssystemen

Hierbei ist *instanzausgangsverzeichnis* das Ausgangsverzeichnis der Instanz und **db2-pfad** die Position, an der Ihre Kopie von DB2 Version 10.1 installiert ist.

6. Wenn Sie für einen DB2-32-Bit-Server vor Version 10.1 ein Upgrade auf einen 64-Bit-Server der Version 10.1 durchgeführt haben, müssen Sie Ihre Textindizes löschen und erneut erstellen. Entsprechende Informationen finden Sie in Kapitel 37, „Löschen eines Textindex“, auf Seite 125 und Kapitel 28, „Erstellen eines Textindex“, auf Seite 97. In Net Search Extender können Sie Textindizes, die Sie in einer 32-Bit-Instanz erstellt haben, nicht in einer 64-Bit-Instanz verwenden. Die Suchmaschine gibt den Fehler CTE0101 mit dem Ursachencode 17 zurück.
7. Wenn Sie Textindizes verwenden möchten, die Sie vor dem DB2-Server-Upgrade unter Windows-Betriebssystemen in den Installationsverzeichnissen Ihrer DB2-Kopien vor Version 10.1 erstellt haben, stellen Sie die Textindexverzeichnisse wieder her,

Entsprechende Informationen finden Sie in Kapitel 39, „Backup und Restore von Textindizes“, auf Seite 129. Die Wiederherstellung der Textindexverzeichnisse ist erforderlich, falls Sie eine DB2-Kopie mit der Upgradeaktion im Fenster **Mit vorhandener Installation arbeiten** während der Installation von DB2 Version 10.1 ausgewählt haben oder Sie Ihre DB2-Kopien einer Version vor Version 10.1 nach dem Upgrade deinstalliert haben.

Die Textindexkonfiguration enthält die Position dieser Textindexverzeichnisse vor der Migration. Abfragen und Indexverwaltungsoperationen, die diese Textindizes verwenden, schlagen fehl, wenn Sie die Textindexverzeichnisse nicht wiederherstellen.

Teil 5. DB2 Net Search Extender-Tool für die Indexmigration von 32-Bit auf 64-Bit

Net Search Extender-Indizes, die unter 32-Bit-DB2-Instanzen erstellt wurden, sind mit 64-Bit-Instanzen nicht kompatibel.

Der Versuch, in einer 64-Bit-DB2-Instanz einen Index zu durchsuchen oder zu aktualisieren, der unter einer 32-Bit-Instanz erstellt wurde, führt immer zu einer Fehlermeldung, die eine Beendigung der Operation verhindert. Ursache ist das Format einer Reihe kleiner, zum Index gehörender Dateien, das eine spezifische Wortgröße aufweist.

Gegenwärtig kann für ein Upgrade von 32-Bit- auf 64-Bit-Instanzen lediglich empfohlen werden, die unter den 32-Bit-Instanzen erstellten NSE-Indizes zu löschen und erneut zu erstellen. Falls Sie Indizes mit geringer oder moderater Größe verwenden, bei denen ein direkter Zugriff auf den Textinhalt einer Tabelle möglich ist, können Sie diese Empfehlung befolgen, ohne dass hierdurch mehr als ein geringer oder moderater technischer Aufwand entsteht. Bei größeren Indizes wird für einen Rebuild der NSE-Indizes wesentlich mehr Zeit benötigt, auch wenn das DB2-Datenbanksystem auf gut optimierten Systemen bis zu 12 GB pro Stunde verarbeitet werden können.

Um einen vollständigen Rebuild der Indizes zu vermeiden, können Sie ein Tool zur Offlinemigration der Indizes verwenden. Die aktuelle Version des Tools unterstützt lediglich AIX- und Solaris-Betriebssysteme.

Bei Content Manager-Systemen befindet sich der durch Net Search Extender indizierte Inhalt für alle nicht auf Attribute bezogenen Indizes nicht lokal in der Tabelle, für die der Index definiert ist. Attribute sind Metadaten, die auf dem Bibliotheksserver gespeichert sind. Ihr Inhalt ist lokal. Für die CM-Systeme sind die Textdokumente für die Net Search Extender-Datenbank, die für die Volltextsuche aktiviert ist, nicht lokal. Stattdessen werden die Dokumente auf einem separaten Dokumentserver gespeichert und zur Indexierung an Net Search Extender gesendet. Der Inhalt wird mit einer erheblichen Latenzabwertung stattdessen aus einer fernen Position abgerufen und gefiltert (bei Content Manager ist dies der Resource Manager). Dies verlangsamt die Indexierung selbst und macht somit die erneute Erstellung von NSE-Indizes äußerst kostenintensiv.

Falls Sie gegenwärtig 32-Bit-Instanzen von DB2 verwenden und DB2 noch mit dem Release von Version 8 ausführen, müssen Sie auf AIX- und Solaris-Plattformen die DB2-Instanzen von 32-Bit auf 64-Bit aktualisieren, bevor Sie eine Migration von Version 8 auf Version 9.1, Version 9.5 oder Version 9.7 vornehmen. Bei all diesen Releases gibt es auf diesen beiden Plattformen keine 32-Bit-Instanzen mehr. Die Situation gilt ebenfalls bei Linux und Windows. Dort werden 64-Bit-Instanzen für umfangreichere System empfohlen, da bei einem 64-Bit-System keine Einschränkungen hinsichtlich der Speicherbelegung bestehen.

Die (im folgenden Abschnitt beschriebene) Indexmigration muss vor einer Migration der DB2-Instanz und -Datenbanken von 32-Bit auf 64-Bit erfolgen.

Indexmigrationsprozedur

1. Laden Sie das Indexmigrationstool von der FTP-Site herunter.

2. Extrahieren Sie die komprimierte Datei in einem beliebigen Verzeichnis auf Ihrem System.
3. Das Indexmigrationstool kann durch die Ausführung des Shell-Scripts **ctemigridx.sh** ausgeführt werden. Führen Sie das Shell-Script (`./ctemigridx.sh`) aus, wie im folgenden Abschnitt beschrieben.

DB2 Version 8 in 32-Bit-AIX-Umgebungen:

1. Melden Sie sich als Instanzeigner an.
2. Extrahieren Sie das Archiv `NSE_32_64_Idx_Migr_Tool_AIX_SOL.tar.gz`.
3. Navigieren Sie zum Verzeichnis `NSE_32_64_Idx_Migr_Tool_AIX_SOL`.
4. Überprüfen Sie, ob der DB2-Datenbankmanager betriebsbereit ist und ob NSE gestoppt wurde.
5. Führen Sie `/usr/sbin/slibclean` aus, um den AIX-Bibliothekscache zu bereinigen. Dies ist notwendig, damit Konflikte zwischen installierten NSE-Bibliotheken und Bibliotheken desselben Namens im Verzeichnis `NSE_32_64_Idx_Migr_Tool_AIX_SOL` verhindert werden. Abhängig von Ihrer Systemkonfiguration benötigen Sie möglicherweise eine Rootberechtigung, um **slibclean** auszuführen.
6. Führen Sie das Shell-Script **ctemigridx.sh** aus. Zur Ausführung des Scripts stehen zwei Modi zur Verfügung.

Beim ersten Modus verwendet das Script einen Datenbanknamen als Argument, ermittelt automatisch alle vorhandenen Indizes in dieser Datenbank und bietet zwei Optionen zur Auswahl, mit denen die Indizes entweder selektiv oder alle auf einmal migriert werden können.

Modus 1: Bei diesem Modus werden nach dem Aufbau einer Datenbankverbindung alle benötigten Daten aus den NSE-Datenbanktabellen abgerufen. Anschließend wird eine Liste der Indizes angezeigt, die für eine Migration bereit sind. Sie können einen Index oder alle aufgelisteten Indizes auswählen. Wenn Sie den Parameter **-check** zum Befehl hinzufügen, werden alle erforderlichen Schritte ohne eine Migration ausgeführt.

Beispiel: `./ctemigridx.sh -dbname sample`

Mit dem Modus 2 kann ein bestimmter Index migriert werden, falls Sie den Indexnamen bereits kennen und eine Migration dieses Indexes beabsichtigen. Dieser Modus ist bei einer großen Anzahl von Indizes hilfreich, denn es entfällt die Notwendigkeit, ein umfangreiches Menü von Indizes durchzuarbeiten. Auch zur Ausführung von wiederholten Tests für einen einzelnen Index ist dieser Modus gut geeignet. Bei diesem Modus werden Indexinformationen als Eingabe verwendet und ohne Benutzerinteraktion automatisch migriert.

Modus 2: `./ctemigridx.sh -i indexname -p indexverzeichnis [-showmap]`

Beispiel: `./ctemigridx.sh -i IX123456 -p /home/user/sqllib/db2ext/indexes`

Das Indexverzeichnis muss genauso angegeben werden, wie es bei der Indexerstellung verwendet wurde. Das Indexverzeichnis enthält immer ein Unterverzeichnis `NODE0000`, in dem sich der eigentliche Index befindet. Falls Sie das Attribut **-showmap** hinzufügen, enthält die Protokolldatei einen Speicherauszug der migrierten Indexattribute (Abschnitte und Attribute), der zu weiteren Prüfzwecken verwendet werden kann.

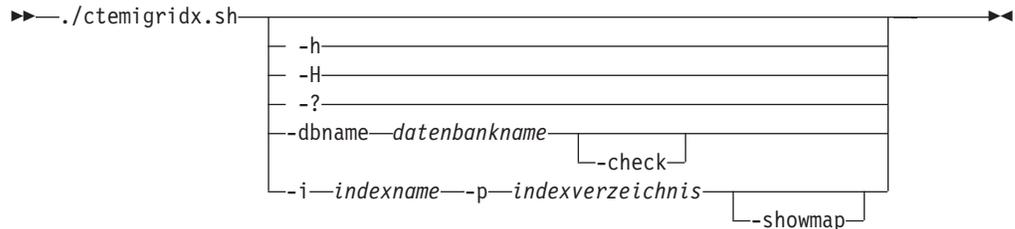
```
. /ctemigridx.sh -i IX123456 -p /home/user/sqllib/db2ext/indexes
-showmap
```

7. Nach dieser Operation ist die Migration abgeschlossen. Der Benutzer kann danach mit der Migration auf einer 64-Bit-Instanz wie in der DB2-Dokumentation beschrieben fortfahren.

DB2 Version 8 in 32-Bit-Solaris-Umgebungen:

Die erforderlichen Schritte sind mit den Schritten identisch, die für AIX ausgeführt werden müssen. Lediglich der Schritt für **slibclean** wird nicht benötigt. **slibclean** ist unter Solaris nicht verfügbar.

Syntax von 'ctemigridx.sh'



-h, -H, -?

Zeigt die Hilfe für den Befehl an und beendet den Befehl.

-dbname datenbankname

Der Name der Datenbank.

-check

Führt zur Simulation aller erforderlichen Schritte ohne eine Migration aus.

-i indexname

Der Name des zu migrierenden Indexes. Er hat stets das Format `IXnnnnnn`.

-p indexverzeichnis

Das Verzeichnis, in dem sich der Index befindet. Es muss genauso angegeben werden, wie es bei der Indexerstellung verwendet wurde. Meistens hat es den Wert `~/sql1lib/db2ext/indexes`.

-showmap

Bewirkt, dass die Protokolldatei einen Speicherauszug der migrierten Indexattribute (Abschnitte und Attribute) enthält, der zu weiteren Prüfzwecken verwendet werden kann.

Anmerkung:

1. Das aktuelle Migrationstool unterstützt die Migration von Indizes nicht, die in einer bereits migrierten Instanz der Version 9 wiederhergestellt wurden.
2. Vor der Ausführung des Indexmigrationstools muss die DB2-Instanz gestartet und NSE gestoppt worden sein.
3. Zur Ausführung des Tools müssen Sie eine Schreibberechtigung für das aktuelle Verzeichnis besitzen.
4. Das Tool führt ein Backup lediglich für die Indexdateien durch, die geändert werden, NICHT jedoch für die gesamten Indizes. Es empfiehlt sich, vor der Ausführung des Tools ein Backup für das gesamte Indexverzeichnis durchzuführen.
5. Sie müssen das Tool als DB2-Instanzeigner ausführen, dürfen die Ausführung jedoch nicht als Root vornehmen.
6. Falls der Befehl `db2_local_ps` keine Ergebnisse liefert, ist DB2 nicht gestartet und das Tool kann nicht ausgeführt werden.
7. Das Migrationstool unterstützt gegenwärtig nur Systeme mit einem einzigen Knoten.

Teil 6. DB2 Net Search Extender-Tool für die Indexmigration von 32-Bit auf 64-Bit (Windows)

DB2 Net Search Extender-Indizes, die unter 32-Bit-DB2-Instanzen erstellt wurden, sind mit 64-Bit-Instanzen nicht kompatibel. Der Versuch, einen Index, der in einer 32-Bit-DB2-Instanz erstellt wurde, in einer 64-Bit-Instanz zu durchsuchen oder zu aktualisieren, führt zur Rückgabe einer Fehlermeldung.

Mit dem Indexmigrationstool wird eine Offlinemigration der Indizes durchgeführt und die erneute Erstellung der Indizes vermieden. Die Migration wirkt sich nur auf eine kleine Anzahl von Metadateien aus und kann schnell und ohne die Übertragung großer Datenmengen durchgeführt werden. Führen Sie für diese Migration das Stapelscript aus, das im Lieferumfang dieses Pakets enthalten ist.

Indexmigration von Version 8 (32-Bit) auf Version 9 (64-Bit)

Dieses Migrationstool kann nur für eine 32-Bit-Quelleninstanz von Net Search Extender ausgeführt werden, die auf eine 64-Bit-Instanz migriert werden soll. Es darf nicht in einer 64-Bit-Instanz ausgeführt werden. Führen Sie ein Backup der Datenbank und des Indexverzeichnisses des V8-Systems durch. Stellen Sie bei der Durchführung des Backups sicher, dass die Metadateien mit den Erweiterungen .an, .as und .tf ordnungsgemäß gesichert werden.

Migrationsschritte

1. Bereiten Sie die Datenbank vor, indem Sie die benutzerdefinierten DATALINKS-Funktionen in der Datenbank von DB2 Version 8 löschen:

```
db2 Drop Specific Function DB2Ext.DataLinkContent1
db2 Drop Specific Function DB2Ext.DataLinkContent1
db2 Drop Specific Function DB2Ext.DataLinkContent2
db2 Drop Specific Function DB2Ext.DataLinkContent4
db2 Drop Specific Function DB2Ext.DataLinkContent3
db2 Disconnect all
```
2. Migrieren Sie die 32-Bit-Indexdateien mithilfe des Migrationstools ctemigridx auf die 64-Bit-Version. Details hierzu finden Sie in der Beschreibung der Ausführung des Tools auf einer V9-32-Bit-Maschine.
3. Führen Sie ein Backup der Datenbank- und Indexdateien der 32-Bit-Instanz durch.
4. Übertragen Sie die Dateien und die Datenbank, für die Sie das Backup durchgeführt haben, auf die Zielmaschine, auf der DB2 Version 9.1 (64-Bit) oder DB2 Version 9.5 (64-Bit) installiert ist.
5. Migrieren Sie die erforderlichen Net Search Extender-Datenbanken auf die Zielmaschine, bevor Sie die Net Search Extender-Indizes auf die Zielmaschine migrieren.
6. Stellen Sie die Net Search Extender-Indizes und die neu generierte 64-Bit-Metadatei auf der Zielmaschine wieder her, indem Sie die folgende Prozedur ausführen:
 - a. Geben Sie den folgenden Befehl zum Stoppen von Net Search Extender ein:

```
db2text stop
```
 - b. Stellen Sie die Backupkopien der Indexverzeichnisse in ihrem ursprünglichen Pfad wieder her.
 - c. Geben Sie den Befehl zum Neustart von Net Search Extender ein.

```
db2text start
```

7. Migrieren Sie die Net Search Extender-Datenbank auf das aktuelle Release.

```
db2extmdb <datenbankname>
```
8. Führen Sie eine Indexänderungsoperation (ALTER INDEX) durch und geben Sie neue Index- und Arbeitsverzeichnisse an. Der Instanzeigner benötigt die entsprechende Berechtigung für den Zugriff auf die Indexdateien und für deren Änderung.

```
db2text alter index <indexname> for text index directory <neues_indexverzeichnis>  
work directory <neues_arbeitsverzeichnis>
```
9. Führen Sie Suchvorgänge wie gewohnt aus und prüfen Sie, ob der neue migrierte Index durchsuchbar ist.

Indexmigration von Version 9 (32-Bit) auf Version 9 (64-Bit)

Dieses Migrationstool kann nur für eine 32-Bit-Quelleninstanz von Net Search Extender ausgeführt werden, die auf eine 64-Bit-Instanz migriert werden soll. Es darf nicht in einer 64-Bit-Instanz ausgeführt werden. Führen Sie ein Backup der Datenbank und des Indexverzeichnisses des V9-32-Bit-Systems durch. Stellen Sie bei der Durchführung des Backups sicher, dass die Metadateien mit den Erweiterungen .an, .as und .tf ordnungsgemäß gesichert werden. Für die Migration von DB2 Version 9 (32-Bit) auf Version 9 (64-Bit) gibt es zwei verschiedene Vorgehensweisen, abhängig davon, ob die Quellen- und die Zielmaschine identisch sind.

- Quellen- und Zielmaschine sind unterschiedlich
 1. Migrieren Sie die aktuelle V9-32-Bit-Datenbank und die Net Search Extender-Indizes mithilfe des Migrationstools `ctemigrdx` auf die 64-Bit-Version. Details hierzu finden Sie in der Beschreibung der Ausführung des Tools auf einer V9-32-Bit-Maschine.
 2. Führen Sie ein Backup der Datenbank- und Indexdateien der 32-Bit-Instanz durch.
 3. Übertragen Sie die Dateien und die Datenbank, für die Sie das Backup durchgeführt haben, auf die Zielmaschine, auf der Version 9 installiert ist.
 4. Migrieren Sie die erforderlichen Net Search Extender-Datenbanken auf DB2 Version 9 (64-Bit), bevor Sie die Net Search Extender-Indizes auf DB2 Version 9 (64-Bit) migrieren.
 5. Stellen Sie die Net Search Extender-Indizes und die neu generierte 64-Bit-Metadatei in der V9-64-Bit-Konfiguration wieder her, indem Sie die folgende Prozedur ausführen:
 - a. Geben Sie den folgenden Befehl zum Stoppen von Net Search Extender ein:

```
db2text stop
```
 - b. Stellen Sie die Backupkopien der Indexverzeichnisse in ihrem ursprünglichen Pfad wieder her.
 - c. Geben Sie den Befehl zum Neustart von Net Search Extender ein.

```
db2text start
```
 6. Migrieren Sie die Net Search Extender-Datenbank auf das aktuelle Release.

```
db2extmdb datenbankname
```
 7. Führen Sie eine Indexänderungsoperation (ALTER INDEX) durch und geben Sie neue Index- und Arbeitsverzeichnisse an. Der Instanzeigner benötigt die entsprechende Berechtigung für den Zugriff auf die Indexdateien und für deren Änderung.

```
db2text alter index indexname for text index directory neues_indexverzeichnis  
work directory neues_arbeitsverzeichnis
```

8. Führen Sie Suchvorgänge wie gewohnt aus und prüfen Sie, ob der neue migrierte Index durchsuchbar ist.
- Quellen- und Zielmaschine sind identisch

Die folgenden Schritte beschreiben die Vorgehensweise bei der Migration einer aktuellen V9-32-Bit-Datenbank und der Indizes auf eine 64-Bit-Windows-Maschine:

1. Migrieren Sie die 32-Bit-Indexdateien mithilfe des Migrationstools `ctemigridx` auf die 64-Bit-Version. Details hierzu finden Sie in der Beschreibung der Ausführung des Tools auf einer V9-32-Bit-Maschine.
2. Migrieren Sie die DB2-Instanz und -Datenbank von der V9-32-Bit-Version auf die V9-64-Bit-Version.
3. Migrieren Sie die Net Search Extender-Datenbank mithilfe des Befehls `db2extmdb` auf das aktuelle Release.

Schritte zur Ausführung des Tools auf einer V9-32-Bit-Maschine:

1. Melden Sie sich als DB2-Instanzeigner an.
2. Extrahieren Sie die Archivdatei `NSE_32_64_Idx_Migr_Tool_WINDOWS.zip` in das Verzeichnis `NSE_32_64_Idx_Migr_Tool_WINDOWS`. Stellen Sie sicher, dass die DB2-Instanz betriebsbereit und Net Search Extender gestoppt ist. Stellen Sie sicher, dass kein anderer Prozess versucht, eine exklusive Sperre für die Indexdateien im Indexverzeichnis zu erhalten.
3. Führen Sie die Stapeldatei `ctemigridx.bat` aus. Das Script kann in zwei verschiedenen Modi ausgeführt werden.

Bei Modus 1 werden nach dem Herstellen einer Datenbankverbindung alle benötigten Daten aus den Net Search Extender-Datenbanktabellen abgerufen. Bei diesem Modus verwendet das Script einen Datenbanknamen als Argument, ermittelt automatisch alle vorhandenen Indizes in dieser Datenbank und bietet zwei Optionen zur Auswahl, mit denen die Indizes entweder selektiv oder alle auf einmal migriert werden können. Anschließend wird eine Liste der Indizes angezeigt, die für eine Migration bereit sind. Sie können einen Index oder alle aufgelisteten Indizes auswählen. Wenn Sie den Parameter `-check` zum Befehl hinzufügen, werden alle erforderlichen Schritte ohne eine Migration ausgeführt.

```
ctemigridx -dbname sample
```

Mit Modus 2 kann ein bestimmter Index migriert werden, falls Sie den Indexnamen bereits kennen und eine Migration dieses Indexes beabsichtigen. Dieser Modus ist bei einer großen Anzahl von Indizes nützlich, denn es entfällt die Notwendigkeit, ein Menü mit zahlreichen Indizes durchzuarbeiten. Darüber hinaus ist dieser Modus nützlich, wenn Sie wiederholte Tests mit einem einzelnen Index durchführen möchten.

```
ctemigridx -i indexkennung -p indexverzeichnis [-showmap]
```

Das Indexverzeichnis muss genauso angegeben werden, wie es bei der Indexerstellung verwendet wurde. Das Indexverzeichnis enthält immer ein Unterverzeichnis `NODE0000`, in dem sich der eigentliche Index befindet. Zurzeit unterstützt das Script nur Einzelknotensysteme. Wenn Sie den Parameter `-showmap` hinzufügen, enthält die Protokolldatei einen Speicherauszug der migrierten Indexattribute für eine zusätzliche Prüfung.

```
ctemigridx -i IX123456 -p D:\sql1ib\db2ext\indexes -showmap
```

4. Nach Abschluss der Net Search Extender-Indexmigration befinden sich einige `*.32`-Dateien im Indexverzeichnis. Den migrierten Indizes entsprechend sind die folgenden neuen Dateien vorhanden:
 - `IX123456.as.32` (Backup der alten 32-Bit-Datei `IX123456.as`)

- IX123456.an.32 (Backup der alten 32-Bit-Datei IX123456.an)
- IX123456.tf.32 (Backup der alten 32-Bit-Datei IX123456.tf)

Bei der Migration wird ein temporäres Verzeichnis erstellt. Ist es nicht vorhanden, befindet sich ein Verzeichnis mit dem Namen TMP_IX123456 im Net Search Extender-Indexverzeichnis.

5. Nach der Prüfung der 64-Bit-Net Search Extender-Indizes können die *.32-Dateien entfernt oder als Backup an einer anderen Position gespeichert werden.

Die Indexmigration ist abgeschlossen. Sie können nun die Migration der Instanz auf die 64-Bit-Version durchführen.

Teil 7. Planungsaspekte

Durch die entsprechende Planung vor der Verwendung von Funktionen können Sie sicherstellen, dass Sie die Datenbankfunktionalität optimal nutzen.

Wichtig: Net Search Extender gilt als veraltet. Die Verwendung des Produkts wird nicht mehr empfohlen; es wird möglicherweise in einem zukünftigen Release entfernt. Verwenden Sie DB2 Text Search als schnelle und vielseitige Methode, um Volltextdokumente, die in DB2-Datenbanken gespeichert sind, mithilfe von SQL- und XQuery-Anweisungen zu durchsuchen. Der Abschnitt zur Migration von Net Search Extender auf DB2 Text Search enthält Details hierzu.

Wenn Sie Net Search Extender möglichst effektiv nutzen möchten, sind einige Planungsschritte vor der Implementierung erforderlich. Die Planung bezieht verschiedene Benutzergruppen mit ein, einschließlich der Datenbankadministratoren, der Schnittstellen- und Systemdesigner, der Systemarchitekten und der Entwickler.

Die folgenden Themen enthalten Hinweise auf die Bereiche, die berücksichtigt werden sollten:

- Verzeichnispositionen und Indexspeicher
- Tabellen-, Spalten- und Indexnamen
- Dokumentformate und unterstützte Codepages
- Vorbeugende Maßnahmen gegen die Beschädigung von Net Search Extender-Indexdateien
- Filtersoftware Outside In
- Benutzeraufgabenbereiche
- Erweiterte textverwaltete Zwischenspeicherinfrastruktur für Teilaktualisierungen

Weitere Informationen zur Entwicklung von auf Net Search Extender basierten Anwendungen finden Sie in folgenden zugehörigen Themen:

Kapitel 13. Verzeichnispositionen und Indexspeicher für Net Search Extender

Der Plattenspeicherbedarf für einen Net Search Extender-Index ist vom Umfang und vom Typ der Daten abhängig, die indexiert werden sollen.

Als Richtlinie für die Indexierung von Einzelbytedokumenten reservieren Sie einen Plattenbereich, dessen Umfang sich durch die Multiplikation der Gesamtgröße der zu indexierenden Dokumente mit dem Faktor 0,7 ermitteln lässt. Für Doppelbytedokumente müssen Sie einen Plattenspeicherbereich reservieren, dessen Größe der Gesamtgröße der zu indexierenden Dokumente entspricht. Die Gesamtgröße muss möglicherweise Daten umfassen, die außerhalb der aktiven Datenbank gespeichert werden, die mithilfe von benutzerdefinierten Funktionen abgerufen werden.

Der Speicherbedarf für temporäre Dateien im Arbeitsverzeichnis liegt zwischen dem 1,0- bis 4,0-fachen des Speicherplatzes, der für die endgültige Indexdatei im Indexverzeichnis erforderlich ist. Das Standardindexverzeichnis ist ein Unterverzeichnis des DB2-Instanzverzeichnisses, das sich normalerweise in der Partition /home des Systems befindet (bei Linux- und UNIX-Betriebssystemen) bzw. im Laufwerk C: (bei Windows-Betriebssystemen). Für den Standardindex können ebenfalls Größenbeschränkungen gelten. Weitere Details finden Sie in „Sichten für Informationen auf Datenbankebene“ auf Seite 285.

Wenn Sie über mehrere große Indizes verfügen, speichern Sie diese auf getrennten Platteneinheiten, insbesondere wenn Sie einen gleichzeitigen Zugriff auf die Indizes bei der Aktualisierung von Indizes oder der Durchführung von Suchoperationen haben.

Für jeden Index sollte sich das entsprechende Index- und Arbeitsverzeichnis in demselben Dateisystem bzw. Laufwerk befinden. Falls Sie nicht die Standardposition verwenden, müssen Sie darauf achten, im Befehl CREATE INDEX beide Positionen (also die Positionen für das Indexverzeichnis und das Arbeitsverzeichnis) so anzugeben, dass sich diese in demselben Dateisystem befinden. Wird lediglich das Indexverzeichnis im Befehl angegeben, führt dies dazu, dass das Arbeitsverzeichnis im Standardpfad, der sich in den meisten Fällen möglicherweise nicht in demselben Dateisystem befindet, erstellt wird (und umgekehrt).

Bei einer partitionierten Datenbank verwendet ein Textindex zum Platzieren des Index immer noch ein einzelnes Dateisystem auf einer physischen Maschine. Bei großen Datenbanken speichern Sie den Index und die Arbeitsverzeichnisse in einem Dateisystem, das sich auf einer RAID-Einheit (RAID = Redundant Array of Independent Disks) befindet. Dadurch verringert sich die Wahrscheinlichkeit, dass beim Verwenden des Textindex ein Engpass bei der Ein-/Ausgabe auftritt.

Verwenden Sie zum Erstellen, Aktualisieren und Löschen von Net Search Extender-Indizes die Befehlszeilenschnittstelle.

Kapitel 14. Überlegungen zu den Ressourcen für partitionierte DB2-Server

Beim Ausführen von NSE in einer Umgebung mit partitionierten Datenbanken sollten Sie folgende Überlegungen zu Ressourcen bedenken:

- Wenn mehrere Textindizes parallel zueinander aktualisiert werden, kann dies während der intensiven Ein-/Ausgabephasen der Indexaktualisierung zu einer beträchtlichen Anzahl von getrennten Plattenbelegungsschritten führen. Dies erfordert sorgfältiges Erwägen bei der Einrichtung der Partitionen, des zugeordneten NSE-Indexspeichers und dem Planen von Indexaktualisierungen.
- Bei NSE-Verwaltungsbefehlen werden zum Ausführen von Operationen wie z. B. dem Erstellen, Löschen oder Aktualisierungen des Index Prozesse für jede Partition gestartet. Der Aktualisierungsprozess kann ein Prozess mit einer längeren Laufzeit sein, der Ressourcen verbraucht. Wenn für geplante Aktualisierungen von Textindizes im System mehrere Indexaktualisierungen gleichzeitig ablaufen, kann dies dazu führen, dass so viele Prozesse pro Partition vorhanden sind, wie Indizes zu einem bestimmten Zeitpunkt aktualisiert werden. Es empfiehlt sich in solch einem Fall, die Anzahl der gleichzeitig ablaufenden Indexaktualisierungen zu verringern, indem der Zeitplan für Indexaktualisierungen entsprechend geplant wird.
- Unter Linux- oder UNIX-Betriebssystemen können Sie den Befehl **ulimit** mit den entsprechenden Optionen verwenden, um die Größe des Verarbeitungssourcengrenzwerts anzuzeigen oder zu ändern. Wenn Sie DB2 Net Search Extender in einer Umgebung mit partitionierten Datenbanken ausführen, verwenden Sie den Befehl **db2_all ulimit** mit den entsprechenden Optionen, um die für alle Datenbankpartitionen gültige Größe anzuzeigen. Die Überprüfung der festen ulimit-Werte des Betriebssystems ist für eine erfolgreiche Ausführung der Net Search Extender-Befehle zur Indexaktualisierung von zentraler Bedeutung. So kann zum Beispiel eine nicht ausreichende Datensegmentgröße in einer der Partitionen zu Fehlern bei den Indexaktualisierungsbefehlen führen; dabei wird normalerweise der Fehler CTE0105 in der Ereignissicht des Textindex protokolliert.

Kapitel 15. Suche mit gespeicherten Prozeduren - Speicherbedarf

Auf verschiedenen Plattformen sind unterschiedliche Speichermengen für die Verwendung des Cache zur Durchführung einer Suche mit gespeicherten Prozeduren erforderlich.

Die Verwendung des Caches für eine Suche mit gespeicherten Prozeduren erfordert große Speicherkapazitäten und stellt für die folgenden Plattformen unterschiedliche Anforderungen an den Speicher:

- AIX
- Windows
- Solaris
- Linux

Net Search Extender-Speicherbedarf für AIX (64-Bit)

Vor der Verwendung von Net Search Extender für AIX (64-Bit) müssen Sie die Systemgrenzwerte, die Grenzwerte für den gemeinsam genutzten Speicher und den Auslagerungsspeicher konfigurieren.

Konfigurieren der Systembegrenzungen:

- Überprüfen Sie die Systembegrenzungen mithilfe des Befehls `ulimit -a`.
- Wenn andere Werte als 'uneingeschränkt' angegeben werden, führen Sie folgende Schritte aus:
 - Melden Sie sich mit Rootberechtigung an.
 - Führen Sie ein Backup der Datei `/etc/security/limits` durch, und editieren Sie die Datei, um die festen Grenzwerte zu erhöhen.
 - Setzen Sie alle Werte auf uneingeschränkt (Wert -1) für den verwendeten DB2-Instanzeigner.

Konfigurieren der Begrenzungen für den gemeinsamen Speicher:

- Unter AIX besteht keine Notwendigkeit, die Begrenzungen für gemeinsamen Speicher zu konfigurieren.

Konfigurieren des Auslagerungsspeichers:

- Rufen Sie die RAM-Größe des Systems mithilfe des Befehls `lsattr -E -l sys0` ab.
- Rufen Sie die Größe des Auslagerungsspeichers (swap space) mithilfe des Befehls `lsp -a` ab.
- Setzen Sie die Größe des Auslagerungsspeichers mindestens auf das 1,5- bis 2fache der RAM-Größe Ihres Systems, oder verwenden Sie den Parameter **MAXIMUM CACHE SIZE** im Befehl **CREATE INDEX**. Wählen Sie den höheren Wert mithilfe des Dienstprogramms SMIT aus.

Speicherbedarf für gespeicherte Prozeduren in Net Search Extender unter Windows (32-Bit und 64-Bit)

Die Größe der Auslagerungsdatei für den virtuellen Speicher muss den Einstellungen der verwendeten Windows-Maschine entsprechend definiert werden.

Anpassen der Größe der Auslagerungsdatei:

- Setzen Sie die Größe der Auslagerungsdatei des virtuellen Windows-Speichers mindestens auf das 1,5- bis 2fache der RAM-Größe Ihres Systems, oder verwenden Sie den Parameter **MAXIMUM CACHE SIZE** im Befehl **CREATE INDEX**. Wählen Sie den höheren Wert aus. Informationen zum Ändern der Größe der Auslagerungsdatei finden Sie in der Windows-Dokumentation.

Bei Windows (32-Bit) ist es empfehlenswert, die maximale Cachegröße von ungefähr 1000 MB (1 GB = 1073741824 Bytes) nicht zu überschreiten.

Net Search Extender-Speicherbedarf für Solaris (64-Bit)

Die Einstellungen für die Systemgrenzwerte, für die Grenzwerte des gemeinsam genutzten Speichers und für den Auslagerungsspeicher müssen überprüft und den Einstellungen der verwendeten Solaris-Maschine entsprechend konfiguriert werden.

Konfigurieren der Systembegrenzungen:

- Überprüfen Sie die Systembegrenzungen mithilfe des Befehls **ulimit -a**.
- Führen Sie anschließend die folgenden Schritte aus:
 - Melden Sie sich mit Rootberechtigung an.
 - Führen Sie ein Backup der Datei `/etc/system` durch, und editieren Sie die Datei, um die festen Grenzwerte zu erhöhen.
 - Prüfen Sie, ob die folgenden Zeilen zumindest auf die angegebenen Minimalwerte gesetzt sind, und fügen Sie sie bei Bedarf hinzu:
`rlim_fd_cur -> Standardwert 64, empfohlener Wert >= 1024`
`rlim_fd_cur_max -> Standardwert 1024, empfohlener Wert >= 4096`

Konfigurieren der Begrenzungen für den gemeinsamen Speicher:

- Prüfen Sie die aktuellen Einstellungen mithilfe des Befehls **sysdef -i**.
- Bearbeiten Sie die Datei `/etc/system`, um die Größe des gemeinsamen Speichers wie folgt zu definieren: `set shmsys:shminfo_shmmax=0xffffffff`
Sie müssen möglicherweise auch die folgenden Parameterwerte erhöhen:
`set shmsys:shminfo_shmmni=512`
`set shmsys:shminfo_shmseg=128` (Starten Sie anschließend das System neu.)

Konfigurieren des Auslagerungsspeichers:

- Rufen Sie die RAM-Größe des Systems mithilfe des Befehls `/usr/sbin/prtconf ab`.
- Überprüfen Sie die Systembegrenzungen mithilfe des Befehls `swap -l`.
- Setzen Sie die Größe des Auslagerungsspeichers mindestens auf das 1,5- bis 2fache der RAM-Größe in Ihrem System, oder verwenden Sie den Parameter **MAXIMUM CACHE SIZE** im Befehl **CREATE INDEX**. Wählen Sie den höheren Wert aus. Informationen zum Hinzufügen von Auslagerungsspeicher finden Sie in der Dokumentation zu Ihrem Solaris-System.
Es ist empfehlenswert, die maximale Cachegröße von ungefähr 2000 MB (2 GB = 2147483647 Bytes) nicht zu überschreiten.

Speicherbedarf für gespeicherte Prozeduren in Net Search Extender unter Linux (32-Bit und 64-Bit)

Die aktuellen Grenzwerte für gemeinsam genutzte Ressourcen und die Systemgrenzwerte müssen für Linux-Betriebssysteme überprüft werden.

In der DB2-Dokumentation finden Sie Informationen über empfohlene Kernelparameter unter Linux.

Der Prüfungsstatus für neue Linux-Kernel und -Distributionen wird häufig aktualisiert. Die neuesten Informationen zu unterstützten Linux-Softwareversionen können Sie unter <http://www.ibm.com/software/data/db2/linux/validate> abrufen.

Mit dem Befehl **ipcs -l** können Sie die aktuellen Grenzwerte für Ihre gemeinsam benutzten Ressourcen anzeigen. Mit dem Befehl **ulimit -a** können Sie die Systemgrenzwerte überprüfen.

Kapitel 16. Aspekte von Tabellen-, Spalten- und Indexnamen

In der Regel ist für alle Tabellen-, Spalten- und Indexnamen die Groß-/Kleinschreibung zu beachten.

Bei Net Search Extender ist es möglich, diese Namen in Groß-/Kleinschreibung anzugeben. Wenn Sie unter Windows Tabellen-, Spalten- und Indexnamen in Groß-/Kleinschreibung angeben möchten, müssen Sie den Namen in einer Zeichenfolge aus umgekehrtem Schrägstrich (\) und Anführungszeichen (") angeben. Beispiel: `\DocTxt\`.

Kapitel 17. Dokumentformate und unterstützte Codepages

Net Search Extender muss das Format (bzw. den Typ) von Textdokumenten kennen, die Sie durchsuchen möchten.

Diese Information wird zur Indexierung von Textdokumenten benötigt.

Net Search Extender unterstützt die folgenden Dokumentformate:

TEXT Textdatei (z. B. ASCII unstrukturiert); im Allgemeinen Text ohne Markup-Formatierung.

HTML
Hypertext Markup Language

XML Extended Markup Language

Das Dokumentformat XML ist der Standard für den Spaltendatentyp XML; es ist das einzige unterstützte Dokumentformat für diesen Datentyp.

GPP Allzweckformat (nur Text mit benutzerdefinierten Formatierungsbefehlen)

Outside In (INSO)

Verwenden Sie dieses Format, wenn Sie Filtersoftware zum Extrahieren von Textinhalten aus PDFs und anderen Textformatierungstools, z. B. Microsoft Word, verwenden.

Für die Dokumentformate HTML, XML, GPP und die Outside In-Filterformate kann das Suchen auf bestimmte Teile eines Dokuments eingeschränkt werden.

Wenn Outside In-Filter nicht verwendet werden können, da das Format Ihres Dokuments nicht unterstützt wird, können Sie eine benutzerdefinierte Funktion schreiben, mit der Sie eine eigene Filterung durchführen können. Diese benutzerdefinierte Funktion muss zum Zeitpunkt der Indexerstellung angegeben werden und dient zur Umsetzung der Daten vom nicht unterstützten Format in ein unterstütztes Format.

Dokumente können indiziert werden, wenn sie mit einer der unterstützten codierten Zeichensatzkennungen (CCSIDs = Coded Character Set Identifiers) gespeichert werden. In der DB2-Dokumentation finden Sie eine Liste dieser Codepages.

Verwenden Sie den folgenden DB2-Befehl, um die Datenbankcodepage zu überprüfen:

```
db2 GET DB CFG for dbname
```

Verwenden Sie hierbei den für die Datenbankcodepage geschriebenen Wert.

Aus Gründen der Einheitlichkeit wandelt DB2 normalerweise die Codepage eines Dokuments in die Codepage der Datenbank um. Wenn Sie jedoch Daten in einer DB2-Datenbank in einer Spalte mit einem Binärdatentyp wie beispielsweise BLOB oder FOR BIT DATA speichern, konvertiert DB2 die Daten nicht, und die Dokumente behalten ihre ursprünglichen CCSIDs.

Beachten Sie, dass durch inkompatible Codepages Probleme auftreten können, wenn ein Textindex erstellt oder eine Suche durchgeführt wird.

Kapitel 18. Vorbeugende Maßnahmen gegen die Beschädigung von Net Search Extender-Indexdateien

Die Beschädigung einer Indexdatei wird durch Fehlermeldungen angezeigt, die in der Ereignistabelle des Index aufgezeichnet werden und Kernelfehler mit verschiedenen Ursachencodes melden, abhängig von den jeweils von der Beschädigung betroffenen Dateien.

Beispiel: Wenn eine Indexaktualisierung versucht wird und der Index beschädigt ist, wird eine Nachricht ähnlich der folgenden in der Ereignissicht angezeigt.

101 CTE0101 Eine Operation der Suchsteuerkomponente ist fehlgeschlagen. Ursache: '7', '100001', '0', 'Rückkehrcode des Kernels: '17'

Ein beschädigter Index kann nicht repariert werden; er muss gelöscht und neu erstellt werden. Um die Beschädigung eines Index zu vermeiden, führen Sie die folgenden vorbeugenden Maßnahmen durch:

- Geben Sie bei Produktionssystemen ein Index- und Arbeitsverzeichnis in der Anweisung CREATE INDEX an und verwenden Sie nicht das Standardverzeichnis für Net Search Extender-Indizes, das sich im Ausgangsverzeichnis des Instanz-eigners befindet. Verwenden Sie für die Index- und Arbeitsverzeichnisse ein separates Dateisystem und überwachen Sie das Dateisystem, um sicherzustellen, dass ausreichend freier Plattenspeicherplatz für Aktualisierungsoperationen zur Verfügung steht. Die für eine Aktualisierung erforderliche Menge an Plattenspeicherplatz ist abhängig von der Größe des Index (besonders von der Größe des Sekundärindex) und von der Menge der während der Aktualisierung zu verarbeitenden Daten. Überwachen Sie die maximale Auslastung während Aktualisierungsoperationen, um die Menge des benötigten Plattenspeicherplatzes zu schätzen.
- Um eine Beschädigung von Indexdateien aufgrund von Windows-Zugriffsverletzungen zu vermeiden, führen Sie die folgenden Schritte aus:
 - Schließen Sie den Textindex und die Arbeitsverzeichnisse aus automatischen Backupprogrammen aus.
 - Schließen Sie die Indexverzeichnisse aus Antivirenprogrammen aus.
 - Inaktivieren Sie den Windows-Indexierservice für die Laufwerke, in denen die Index- und Arbeitsdateien gespeichert sind.
- Stoppen Sie vor dem Systemabschluss Net Search Extender ordnungsgemäß mit den folgenden Befehlen:
 - `db2text control list all locks for database datenbankname`. Wiederholen Sie diesen Befehl, bis keine Sperren mehr aktiv sind. Weitere Informationen hierzu finden Sie in der Beschreibung zum „Befehl UPDATE INDEX“ auf Seite 247.
 - `db2text stop`

Kapitel 19. Filtersoftware Outside In

Net Search Extender unterstützt die Dokumentfiltersoftware eines Drittherstellers.

Diese als Outside In Transformation Technology von Oracle vertriebene Software kann zum Extrahieren von Textinhalten aus PDF-Dateien oder aus Dokumenten verwendet werden, die mit einem der gängigen Textformatierungstools im jeweils zugehörigen Format geschrieben wurden. Der Einsatz nativer Anwendungen ist hierbei nicht erforderlich. Die Beispielformate umfassen z. B. Microsoft Word und Lotus Word Pro.

Die Outside In-Bibliotheken werden während der Indexaktualisierung (**UPDATE INDEX**) von Net Search Extender als Plug-ins geladen. Die Bibliotheken sind nicht Teil von Net Search Extender und müssen separat installiert werden. Dabei muss sichergestellt werden, dass Net Search Extender die Outside In-Bibliotheken lokalisieren kann.

Die Outside In-Software generiert neben Textinhalten auch Strukturinformationen, z. B. Felder. Mit Net Search Extender kann außerdem angepasst werden, welcher Teil der mit Outside In generierten Dokumentinformationen im Index gespeichert werden soll. Dazu muss ein bestimmter Dokumentmodelltyp angewendet werden, das Outside In-Dokumentmodell.

Eine Liste der Filterformate und unterstützten Plattformen finden Sie auf der Website von Oracle unter <http://www.oracle.com>.

Kapitel 20. Benutzeraufgabenbereiche

Zu den Benutzeraufgabenbereichen gehören DB2-Instanzeigner, Datenbankadministratoren und Eigner von Texttabellen; diese verfügen jeweils über eine bestimmte Gruppe von Verwaltungsberechtigungen.

DB2-Instanzeigner

Der DB2-Instanzeigner kann die Instanzservices für DB2 Net Search Extender starten und stoppen sowie die Sperrservices steuern. Darüber hinaus wird dem DB2-Instanzbenutzer für jede aktivierte Datenbank die Berechtigung DBADM erteilt. Dadurch können alle von Net Search Extender getätigten Änderungen über einen zentralen Steuerungspunkt verwaltet werden.

Erforderliche DB2-Berechtigungen

SECADM muss dem Instanzeigner die Berechtigung DBADM mit dem Zugriffsrecht DATAACCESS erteilen. Diese Berechtigungen sind Voraussetzung für die Ausführung von DB2 Net Search Extender-Verwaltungsbefehlen.

Erforderliche Berechtigungen für das Dateisystem

Schreib- und Lesezugriff für alle Textindexverzeichnisse, Lesezugriff auf Modelldateien.

Befehle für den Instanzeigner

DB2TEXT START, DB2TEXT STOP, DB2TEXT CONTROL und DB2EXTHL

Die Befehle sind nur auf dem Server zulässig. In einer Umgebung mit partitionierten Datenbanken kann hierbei jeder beliebige konfigurierte Knoten eingesetzt werden. Jeder Befehl prüft, ob der Benutzer, der den Befehl ausführt, der DB2-Instanzeigner ist. Wenn Sie sich als Instanzeigner dazu entschließen, eine abgeschirmte Benutzer-ID zum Ausführen der gespeicherten Prozedur und der benutzerdefinierten Funktionen zu verwenden, muss der abgeschirmte Benutzer über Lese- und Schreibzugriff auf alle Dateien im Indexverzeichnis verfügen (und über Lesezugriff auf den gesamten Verzeichnispfad). Bitte beachten Sie, dass die ID des abgeschirmten Benutzers und die ID des Instanzbenutzers derselben Primärgruppe angehören müssen, damit die Instanzbenutzer-ID über ordnungsgemäßen Zugriff auf die Dateien verfügt, die von der ID des abgeschirmten Benutzers erstellt wurden, und umgekehrt. Ordnen Sie die richtige Gruppenzugehörigkeit und die richtigen Dateiberechtigungen zu.

Neben dem Instanzeigner ist jeder Benutzer, der zu derselben Primärgruppe wie der Instanzeigner gehört, in der Lage, die Befehle **DB2TEXT START, DB2TEXT STOP, DB2TEXT CONTROL und DB2EXTHL** auszuführen.

Datenbankadministratoren

Datenbankadministratoren können Datenbanken zur Verwendung mit Net Search Extender aktivieren und inaktivieren.

Erforderliche DB2-Berechtigungen

DBADM

Befehle für den Datenbankadministrator

DB2TEXT ENABLE DATABASE und DB2TEXT DISABLE DATABASE.

Texttabelleneigner

Der Texttabelleneigner kann Indizes erstellen, löschen und ändern. Beachten Sie, dass Texttabelleneigner in der Lage sein müssen, die Position von Indizes sowie Aktualisierungen an den Volltextindizes zu steuern (über Les- und Schreibzugriff).

Erforderliche DB2-Berechtigungen und -Zugriffsrechte

Eigner der Texttabelle.

Befehle für den Texttabelleneigner:

DB2TEXT CREATE INDEX, DB2TEXT DROP INDEX, DB2TEXT ALTER INDEX, DB2TEXT ACTIVATE CACHE, DB2TEXT DEACTIVATE CACHE, DB2TEXT UPDATE INDEX, DB2TEXT CLEAR EVENTS und DB2EXTTH.

Beachten Sie, dass die Befehlsimplementierung zum Teil unter der Benutzer-ID des DB2-Instanzeigners ausgeführt wird. Aus diesem Grund muss dem Instanzeigner vor der Erstellung oder Änderung der Textindizes der erforderliche Dateisystemzugriff erteilt werden. Eine Auflistung der für jeden Befehl erforderlichen Berechtigungen finden Sie in Kapitel 57, „Verwaltungsbefehle für den Texttabelleneigner“, auf Seite 219.

Kapitel 21. Erweiterte textverwaltete Zwischenspeicherinfrastruktur für Teilaktualisierungen

In Version 9.7 ist eine Konfigurationsoption verfügbar, mit der eine Zwischenspeicherinfrastruktur hinzugefügt werden kann, die das Erfassen von Änderungen ermöglicht, die über die Trigger nicht in der regulären Protokolltabelle erkannt werden.

In Version 9.7 ist eine Konfigurationsoption verfügbar, mit der eine Zwischenspeicherinfrastruktur hinzugefügt werden kann, die das Erfassen von Änderungen ermöglicht, die über die Trigger nicht in der regulären Protokolltabelle erkannt werden. Wenn diese Option aktiviert ist, werden Aktualisierungen über einen Trigger in der regulären Protokolltabelle erfasst und Einfügungen und Löschungen werden in der textverwalteten Zwischenspeichertabelle erfasst.

Diese Konfigurationsoption ist für bereichspartitionierte Tabellen standardmäßig aktiviert und für nicht partitionierte Tabellen inaktiviert. Das Hinzufügen der textverwalteten Zwischenspeicherinfrastruktur hat bei verschiedenen Datenbankoperationen einen kritischen Einfluss auf die Verfügbarkeit und den Status der Basistabelle.

Die Auswirkung der textverwalteten Zwischenspeicherinfrastruktur ähnelt der Auswirkung beim Hinzufügen einer MQT (MQT - Materialized Query Table) mit verzögerter Aktualisierung. Auch wenn die textverwaltete Infrastruktur keine Daten in einer MQT verwaltet, verursacht die Zwischenspeichertabelle ein ähnliches Verhalten wie die MQT-Zwischenspeichertabelle.

Die Tabellen erfordern z. B. nach einer LOAD-Einfügung die Integritätsverarbeitung, um nachfolgende Datenbankoperationen für die Basistabelle zu ermöglichen.

Wenn die Tabellen nur mit Datenbankbefehlen aktualisiert werden, die alle Zeilen der Tabelle betreffen, wie z. B. mit dem Befehl **LOAD REPLACE**, bietet das Hinzufügen der erweiterten Zwischenspeicherinfrastruktur keinen Nutzen; es sollte stattdessen der Index erneut erstellt werden.

Teil 8. Net Search Extender - Verwaltung

Kapitel 22. Net Search Extender-Instanzservices

DB2 Net Search Extender-Instanzservices umfassen Sperren- und Aktualisierungsservices.

Die DB2 Net Search Extender-Instanzservices bestehen aus folgenden Services:

- Sperrenservices
- Aktualisierungsservices

DB2 Net Search Extender-Instanzservices unter Windows werden durch Windows-Dienste dargestellt. Auf einer nicht partitionierten DB2-Instanz steht für die DB2-Instanz ein einziger Service dieser Art mit dem folgenden Servicenamen zur Verfügung:

DB2EXT - *instanzname*

Auf einer partitionierten DB2-Instanz steht pro Partition der DB2-Instanz ein solcher Service mit dem folgenden Servicenamen zur Verfügung:

DB2EXT - *<instanzname>[-<knotennummer>]*

In den folgenden Themen wird das Starten und Stoppen der DB2 Net Search Extender-Instanzservices erläutert. Dabei wird auf die Sperrenservices und die Aktualisierungsservices detailliert eingegangen:

- Starten und Stoppen der NSE-Instanzservices
- Sperrenservices
- Aktualisierungsservices
- NSE-Informationskataloge

Kapitel 23. Starten und Stoppen der Net Search Extender-Instanzservices mithilfe der Befehlszeile

Bevor Sie Textindizes pflegen und Ihre Dokumente durchsuchen können, müssen Sie die Net Search Extender-Instanzservices starten.

Informationen zu diesem Vorgang

Bei DB2-Instanzen, die mit partitionierten Datenbanken verwendet werden, empfiehlt es sich sehr, dass die Net Search Extender-Instanzservices mit den Befehlen **db2text start** bzw. **db2text stop** anstatt mithilfe der normalen Windows-Methoden gestartet und gestoppt werden. Dadurch wird sichergestellt, dass die Instanzservices in ordnungsgemäßer Reihenfolge gestartet und gestoppt werden.

Anmerkung:

- Pro DB2-Instanz muss ein Net Search Extender-Instanzservice vorhanden sein. Der Sperrenservice verwaltet die Sperren für alle aktivierten Datenbanken für diese Instanz.
- DB2 Net Search Extender-Instanzservices unter Windows werden durch Windows-Dienste dargestellt. Auf einer partitionierten DB2-Instanz steht für jede Partition der DB2-Instanz ein solcher Service zur Verfügung.

Vorgehensweise

- Melden Sie sich zum Starten der Instanzservices mit der Benutzer-ID des DB2-Instanzeigners (nur bei UNIX-Betriebssystemen) an, und geben Sie folgenden Befehl ein:
`db2text start`
- Geben Sie zum Stoppen der Instanzservices folgenden Befehl ein:
`db2text stop`

Kapitel 24. Net Search Extender-Sperrenservices

Mit Net Search Extender-Sperrenservices wird verhindert, dass sich Lese- und Schreibprozesse gegenseitig beeinträchtigen.

Wenn Sie Net Search Extender starten, werden auch die Sperrenservices automatisch gestartet. Die Sperrenservices werden benötigt, um den gleichzeitigen Zugriff auf Textindizes in Net Search Extender zu synchronisieren.

Die Sperrenservices stellen sicher, dass nicht zwei Prozesse versuchen, gleichzeitig einen Textindex zu ändern, oder dass nicht ein Prozess Textindexdaten liest, während ein anderer Prozess Änderungen an den gleichen Textindexdaten vornimmt. Aus diesem Grund fordern die meisten Prozesse vor dem Starten eine Sperre für einen Textindex an und geben die Sperre wieder frei, wenn die Verarbeitung abgeschlossen ist.

Beachten Sie, dass die Sperrenservices für Net Search Extender-Textindizes nicht mit DB2-Sperren verwechselt werden dürfen, die den Zugriff auf DB2-Tabellen steuern.

Verwenden der Sperrenservices

In Net Search Extender werden verschiedene Sperrentypen verwendet, die den gleichzeitigen Zugriff auf einen Index steuern.

Die Verwendung der verschiedenen Sperren ist davon abhängig, ob der Textindex nur gelesen wird, wie dies z. B. bei einer Suchanforderung der Fall ist, oder ob Änderungen am Textindex berechnet und anschließend auch in den zugehörigen Dateien nachvollzogen werden müssen; dies ist der Fall bei einer Indexaktualisierung.

Während der Ausführung des Befehls **db2text start** werden die Sperrenservices automatisch gestartet. Es gibt folgende Typen von Sperren für einen Textindex:

S-Sperre

Für den gemeinsamen Lesezugriff, z. B. bei Suchanforderungen.

U-Sperre

Für den Schreib-/Lesezugriff während der Berechnung von Indexänderungen (Aktualisierungen) bei gleichzeitigem Lesezugriff.

X-Sperre

Für den exklusiven Schreib-/Lesezugriff für einen kurzen Zeitraum, in dem Änderungen tatsächlich in den Index geschrieben werden.

IX-Sperre

Für den beabsichtigten Schreib-/Lesezugriff, um neue S-Sperren zu verhindern, während der Aktualisierungsprozess auf eine X-Sperre wartet.

Pro DB2-Instanz ist ein Net Search Extender-Sperrenservice vorhanden. Der Sperrenservice verwaltet die Sperren für mehrere Datenbanken.

Die Konfigurationsdatei für die Sperrenservices hat den Namen `db2ext1m.cfg`. Sie ist auf UNIX-Systemen unter `ausgangsverzeichnis_des_instanzeignerse/sql1lib/db2ext` und auf Windows-Systemen unter `sql1lib\DB2INSTANCE\db2ext` gespeichert.

Änderungen der Konfigurationsdatei werden erst dann wirksam, wenn die Net Search Extender-Instanzservices mit **db2text start** gestartet werden. Der Benutzer kann die folgenden Werte definieren:

- Die maximale Anzahl von Datenbanken
- Die maximale Anzahl von Indizes pro Datenbank
- Die maximale Anzahl zulässiger Sperren (gleichzeitige Benutzer) pro Index
- Wartezeiten und die Anzahl von Versuchen, eine Sperre zu aktivieren

Die Standardwerte für die Konfigurationsdatei lauten wie folgt:

```
<default
    maxDbs          = " 8"
    maxIdxPerDb     = " 50"
    maxLocksPerIdx = "100"

    sWait = " 50"
    uWait = " 500"
    xWait = " 500"

    sAttempt = "50"
    uAttempt = "10"
    xAttempt = "60"

    latchTimeout = "80"

/>
```

Die Syntax lautet `<standardattribut=wert.../>`, wobei die Attribute folgende Bedeutungen haben:

maxDbs

Die Anzahl der Datenbanken, die von den Sperrenservices verwaltet werden können (integer >1).

maxIdxPerDb

Die Anzahl der Indizes pro Datenbank, die gesperrt werden können (integer >1). Dieser Wert ist für alle Datenbanken gleich.

maxLocksPerIdx

Die Anzahl der Sperren, die gleichzeitig für einen Index vorhanden sein können (integer >1). Dieser Wert ist für alle Indizes gleich.

Die Verwendung von gemeinsam genutztem Speicher verhält sich proportional zu dem Produkt der oben genannten drei Maximalwerte ('max...'). Um die übermäßige Verwendung von gemeinsam genutztem Speicher zu vermeiden, stellen Sie sicher, dass die verwendeten Werte mit der tatsächlichen Konfiguration Ihrer DB2-Instanz übereinstimmen. Wenn Sie die Werte für maxDbs, maxIdxPerDb bzw. maxLocksPerIdx so erhöhen, dass sie über den Standardwerten in der oben genannten Konfigurationsdatei liegen, müssen Sie sicherstellen, dass Sie über ausreichend Speicher verfügen. Beachten Sie bei Verwendung einer Umgebung mit partitionierten Datenbanken insbesondere die Werte für 'maxIdxPerDb' und 'maxLocksPerIdx', da diese Einstellungen für jede Partition verwendet werden. Dies ist besonders beim Ermitteln des Speicherbedarfs wichtig, wenn für eine DB2-Instanz auf einer physischen Maschine eine Reihe von logischen Partitionen definiert wird.

sWait/sAttempt

Bei der Anforderung einer S-Sperre ist 'sAttempt' die Anzahl der unternommenen Versuche, falls die Sperre nicht sofort erteilt wird. 'sWait' ist die Wartezeit zwischen diesen Versuchen (integer >1). Diese Parameter gelten auch für IX-Sperren.

uWait/uAttempt

Bei der Anforderung einer U-Sperre ist 'uAttempt' die Anzahl der unternommenen Versuche, falls die Sperre nicht sofort erteilt wird. 'uWait' ist die Wartezeit zwischen diesen Versuchen (integer >1).

xWait/xAttempt

Bei der Anforderung einer X-Sperre ist 'xAttempt' die Anzahl der unternommenen Versuche, falls die Sperre nicht sofort erteilt wird. 'xWait' ist die Wartezeit zwischen diesen Versuchen (integer >1).

latchTimeout

Dies ist die zusätzliche Wartezeit für die Intervallsperrenservices. Ermitteln Sie die Gesamtwartezeit für eine Sperre unter Verwendung folgender Berechnung:

$$\text{Wartezeit} = \# \text{ Versuche} * (\# \text{ Wartezeiten} + (2 * \# \text{ latchTimeout}))$$

Es empfiehlt sich sehr, die Standardwerte für die Parameter für Wartestatus, Versuch und Zeitlimitüberschreitung nicht zu ändern. Die Wartezeit wird in Millisekunden berechnet. Beachten Sie, dass mit jedem Versuch der Wert für latchTimeout verdoppelt wird, wenn er zur Gesamtwartezeit hinzuaddiert wird.

Anzeigen einer Sperrenmomentaufnahme

Zum Anzeigen der Sperrenmomentaufnahme können verschiedene Befehle verwendet werden. Beim erstmaligen Sperren eines Textindex wird sowohl für die Datenbank als auch für den Textindex Speicherplatz in den Sperrenservices reserviert.

Informationen zu diesem Vorgang

Sie können sich eine Sperrenmomentaufnahme ansehen, indem Sie einen der folgenden Befehle verwenden:

- Für einen einzelnen Textindex:
`db2text CONTROL LIST ALL LOCKS FOR DATABASE meinedatenbank INDEX meinindex`

- Für alle gesperrten Textindizes einer Datenbank:

```
db2text CONTROL LIST ALL LOCKS FOR DATABASE meinedatenbank
```

Beachten Sie hierbei, dass nur tatsächlich gesperrte Indizes in der Liste aufgeführt sind.

Beim erstmaligen Sperren eines Textindex wird sowohl für die Datenbank als auch für den Textindex Speicherplatz in den Sperrenservices reserviert. Werden weitere Textindizes gesperrt, wird für diese Sperren ebenfalls Speicherplatz in den Sperrenservices zugeordnet. Dieser Speicher wird erst dann wieder freigegeben, wenn der Textindex gelöscht oder die Datenbank inaktiviert wird oder wenn die Net Search Extender-Services erneut gestartet werden. Dies bedeutet, dass ein Textindex oder eine Datenbank Speicherplatz in den Sperrenservices einnimmt, selbst wenn momentan keine Sperren aktiviert sind.

Der Befehl "db2text CONTROL CLEAR ALL LOCKS" erzwingt die Freigabe aller Sperren für eine Datenbank oder einen Index. Einzelheiten zur Verwendung dieses Befehls finden Sie in „Befehl CONTROL“ auf Seite 207. Verwenden Sie stets die Indexkennung, wenn Sie den Befehl **CLEAR ALL LOCKS** verwenden. Verwenden Sie diesen Befehl nur, nachdem Sie sorgfältig überprüft haben, dass für den Index, für den Sie die Sperren aufheben wollen, keine Indexaktualisierung aktiv ist. Das Aufheben von Sperren für einen Index, der zurzeit aktualisiert wird, kann zur Beschädigung des Index führen und eine vollständige Wiederherstellung des Index erfordern. Beachten Sie, dass dieser Befehl keinen Speicherplatz freigibt, der der Datenbank oder den Indizes zugeordnet ist. Um Speicher freizugeben, müssen Sie entweder den Index löschen, die Datenbank inaktivieren oder die Net Search Extender-Services erneut starten. Geben Sie keine Sperren während eines aktiven Indexaktualisierungsprozesses frei.

Kapitel 25. Aktualisierungsservices

Tabellenänderungen und Indexaktualisierungen sind nicht synchron. Der Indexaktualisierungsprozess kann manuell oder zu bestimmten Zeiten automatisch nach einem Zeitplans gestartet werden.

Die Aktualisierungsservices stellen diese Funktion bereit und werden während des Befehls **db2text start** gestartet.

Bei der Indexerstellung können Sie mit folgendem Befehl angeben, wie oft die Aktualisierungsservices prüfen sollen, ob eine Aktualisierung des Index erforderlich ist:

```
db2text create index DB2EXT.TITLE for text on DB2EXT.TEXTTAB (TITLE)
        UPDATE FREQUENCY D(1,3) H(0,12) M(0) update minimum 5
```

In diesem Beispiel bedeutet dies, dass die Aktualisierungsservices jeden Montag und Mittwoch um 12:00 Uhr und um 0:00 Uhr aktiv werden und prüfen, ob für den Index `db2ext.title` auszuführende Arbeiten aufgelaufen sind. Beachten Sie, dass in diesem Beispiel mindestens fünf Änderungen für `DB2EXT.TITLE` vorliegen müssen, bevor die automatische Indexaktualisierung die Synchronisierung des Textindex mit der Datenbank startet.

In einer Umgebung mit partitionierten Datenbanken werden für jeden der Knoten getrennte Aktualisierungsservices gestartet. Wenn mehrere Textindizes parallel zueinander aktualisiert werden, kann dies während der intensiven Ein-/Ausgabephasen der Indexaktualisierung zu einer beträchtlichen Anzahl von getrennten Plattenbelegungsschritten führen. Dies erfordert bei der Planung und Ausführung von Indexaktualisierungen sorgfältige Beachtung.

Anmerkung:

Wenn der Indexaktualisierungsprozess in zu kurzen Zeitabständen stattfindet, wird die Systemleistung beeinträchtigt. Sie müssen die Menge der Änderungen in Betracht ziehen, von der Sie erwarten, dass sie während jeder Aktualisierung verarbeitet wird; beachten Sie auch die Menge an Zeit, die dafür erforderlich ist, sowie die Anzahl an Indizes, die Sie während der automatischen Indexaktualisierung verarbeiten möchten. Stellen Sie sicher, dass die Intervalle zwischen den einzelnen Indexaktualisierungen groß genug sind, damit eine Aktualisierung fertig gestellt werden kann, bevor die nächste geplante Aktualisierung beginnt, und dass Aktualisierungen für mehrere Indizes nicht gleichzeitig gestartet werden.

Wenn die textverwaltete Zwischenspeicherinfrastruktur für einen Textindex konfiguriert ist, stellen Sie sicher, dass sich die Zwischenspeichertabelle nicht in anstehendem Modus befindet, indem Sie den Befehl **RESET PENDING** ausführen.

Teil 9. Entwicklung: Erstellen und Pflegen eines Textindex

Zu bestimmten Bereichen der Erstellung und Pflege von Textindizes sind wichtige Informationen verfügbar. Lesen Sie diese Informationen, bevor Sie die Ausführung fortsetzen.

Wichtig: Net Search Extender gilt als veraltet. Die Verwendung des Produkts wird nicht mehr empfohlen; es wird möglicherweise in einem zukünftigen Release entfernt. Verwenden Sie DB2 Text Search als schnelle und vielseitige Methode, um Volltextdokumente, die in DB2-Datenbanken gespeichert sind, mithilfe von SQL- und XQuery-Anweisungen zu durchsuchen. Der Abschnitt zur Migration von Net Search Extender auf DB2 Text Search enthält Details hierzu.

Dieser Abschnitt enthält Informationen zur Erstellung und Pflege eines Textindex und umfasst die folgenden Bereiche:

- Einführung in die **db2text**-Befehle
- Aktivieren einer Datenbank für die Textsuche
- Erstellen eines Textindex für verschiedene Datentypen
- Erstellen eines Textindex für einen Kurznamen mit Indexteilaktualisierung unter Verwendung der DB2-Replikation
- Erstellen eines Textindex zur Verwendung durch die Suche mit gespeicherten Prozeduren
- Textindizes für Sichten
- Pflegen eines Index
- Erstellen eines Textindex für eine bereichspartitionierte Tabelle

Darüber hinaus finden Sie in diesem Abschnitt Informationen zur Vermeidung von Codepageproblemen, die möglicherweise auftreten, sowie zu Leistungsaspekten, die zu berücksichtigen sind.

Vergewissern Sie sich vor dem Erstellen eines Index, dass die in Teil 7, „Planungsaspekte“, auf Seite 57 aufgeführten Voraussetzungen erfüllt wurden. Stellen Sie auch sicher, dass Sie die Net Search Extender-Instanzservices mithilfe des Befehls **db2text start** gestartet haben.

Kapitel 26. Aktivieren einer Datenbank

Der Befehl **ENABLE DATABASE FOR TEXT** bereitet die Datenbank für die Verwendung durch Net Search Extender vor.

Vorbereitende Schritte

Die Berechtigung DBADM ist erforderlich.

Informationen zu diesem Vorgang

Führen Sie diese Aktion ein Mal für jede Datenbank aus, die Tabellen mit Spalten mit zu durchsuchendem Text enthält.

Der Befehl **ENABLE DATABASE FOR TEXT** registriert außerdem die Net Search Extender-Suchfunktionen und Prozeduren, die in Kapitel 60, „Skalare SQL-Suchfunktion und SQL-Tabellenwertfunktion“, auf Seite 267 beschrieben sind.

Durch die Aktivierung einer Datenbank erstellt der Befehl außerdem die folgenden Tabellen und Sichten automatisch:

db2ext.dbdefaults

Speichert die Datenbankstandardwerte für Index-, Text- und Verarbeitungsmerkmale.

db2ext.textindexformats

Speichert die Liste der unterstützten Formate und der momentan aktiven verwendeten Modelldateien.

db2ext.indexconfiguration

Speichert die Indexkonfigurationsparameter.

db2ext.textindexes

Eine Katalogsicht, in der alle Textindizes aufgezeichnet werden.

Wenn eine Datenbank aktiviert wurde, bleibt sie aktiviert, bis Sie sie wieder inaktivieren.

Kapitel 27. Inaktivieren einer Datenbank

Wenn Sie beabsichtigen, keine Textsuchvorgänge mehr in einer Datenbank durchzuführen, können Sie die Datenbank mit dem Befehl **DISABLE DATABASE FOR TEXT** inaktivieren.

Vorbereitende Schritte

Die Berechtigung DBADM ist für die Datenbank erforderlich.

Informationen zu diesem Vorgang

Bei der Vorbereitung der Datenbank für die Verwendung durch Net Search Extender werden gewisse verwaltungstechnische Änderungen durchgeführt. Dieser Abschnitt beschreibt die Funktionen, die Ihnen bei der Rücknahme dieses Prozesses helfen.

Verwenden Sie zur Inaktivierung des verbundenen Subsystems folgenden Befehl:
db2text DISABLE DATABASE FOR TEXT

Wenn Sie eine Datenbank inaktivieren, werden mit dem Befehl die folgenden Objekte und auch alle vorhandenen Textindizes gelöscht:

- Die Net Search Extender-Katalogsichten und -Tabellen, die bei der Aktivierung des Servers erstellt wurden.
- Die Deklaration der SQL-Funktionen (UDFs) von Net Search Extender.

Verwenden Sie den folgenden Befehl, wenn der Befehl **DISABLE DATABASE FOR TEXT** einen Fehler zurückgibt, Sie aber unter allen Umständen die Inaktivierung durchführen möchten (selbst dann, wenn Indizes noch verwendet werden):

```
db2text DISABLE DATABASE for text force
```

Anmerkung: Das Inaktivieren einer Datenbank schlägt fehl, wenn in der Datenbank Textindizes definiert sind. Sie sollten diese Indizes einzeln entfernen und dann prüfen, ob irgendwelche Fehler auftreten. Wenn Sie den Befehl **DISABLE DATABASE FOR TEXT FORCE** verwenden, wird nur garantiert, dass Net Search Extender-Katalogtabellen in der Datenbank entfernt werden. Die Option FORCE kann nicht angewendet werden, wenn ein vorhandener Textindex die textverwaltete Zwischenspeicherinfrastruktur verwendet.

Können einige der Indizes jedoch nicht vollständig gelöscht werden, sind möglicherweise immer noch Ressourcen vorhanden, die manuell bereinigt werden müssen. Hierzu gehören z. B. die folgenden Komponenten:

- Dateien im Index-, Arbeits- und Cacheverzeichnis.
- Schemaeinträge in der Datei ctedem.dat.
- Wenn ein Index mit der Replikationserfassungsoption erstellt wurde, müssen die Einträge IBMSNAP_SIGNAL, IBMSNAP_PRUNE_SET und IBMSNAP_PRUNCNTL in den Tabellen der fernen Datenbank manuell gelöscht werden. Diese Einträge können mit der Bedingung APPLY_QUAL='NSEDB2' || *instanzname* and TARGET_SERVER= *datenbankname* einfach identifiziert werden.

Im folgenden Beispiel wird als Instanz DB2 und als Datenbank SAMPLE verwendet.

```
DELETE FROM <ccSchema>.IBMSNAP_SIGNAL
WHERE SIGNAL_INPUT_IN IN
      (SELECT MAP_ID FROM <ccSchema>.IBMSNAP_PRUNCNTL
       WHERE APPLY_QUAL= 'NSEDDB' AND TARGET_SERVER= 'SAMPLE');
```

```
DELETE FROM <ccSchema>.IBMSNAP_PRUNCNTL
WHERE APPLY_QUAL= 'NSEDDB2' AND TARGET_SERVER= 'SAMPLE';
```

```
DELETE FROM <ccschema>.IBMSNAP_PRUNE_SET
WHERE APPLY_QUAL= 'NSEDDB2' AND TARGET_SERVER= 'SAMPLE';
```

Kapitel 28. Erstellen eines Textindex

Setzen Sie den Befehl **CREATE INDEX FOR TEXT** ein Mal für jede Spalte ab, die zu suchenden Text enthält.

Vorbereitende Schritte

Eine der folgenden Berechtigungsstufen ist erforderlich:

- Zugriffsrecht **CONTROL** für die Indextabelle
- Zugriffsrecht **INDEX** für die Tabelle und die Berechtigung **IMPLICIT_SCHEMA** für die Datenbank oder das Zugriffsrecht **CREATEIN** für ein Indexschema
- Berechtigung **DBADM**

Informationen zu diesem Vorgang

Sie können einen Textindex für alle Datentypen erstellen, obwohl für die folgenden Datentypen andere Voraussetzungen gelten:

- Binäre Datentypen
- Nicht unterstützte Datentypen

Für die Erstellung eines Textindex für Suchen mit gespeicherten Prozeduren gelten ebenfalls andere Voraussetzungen.

Wenn Sie einen Textindex erstellen, werden dabei von Net Search Extender automatisch folgende Objekte erstellt, je nachdem, ob die erweiterte textverwaltete Zwischenspeicherinfrastruktur für den Textindex aktiviert wurde oder nicht.

Anmerkung: Die Angabe der Klausel **ADMINISTRATION TABLES IN** ist obligatorisch, wenn ein Index für eine bereichspartitionierte Tabelle erstellt wird. Weitere Informationen hierzu finden Sie im Abschnitt CTE0150E.

- Mit der regulären Protokollinfrastruktur:

Protokolltabelle

Dient der Aufzeichnung aller Änderungen an Zeilen in der Benutzertabelle. Beachten Sie, dass die Protokolltabelle nicht erstellt wird, wenn Sie die Option **RECREATE INDEX ON UPDATE** auswählen oder Replikationserfassungstabellen verwenden.

Ereignistabelle

Dient zum Sammeln von Informationen zu allen Aktualisierungen und möglichen Problemen während einer Aktualisierung der Textindizes.

Trigger für die Benutzertabelle (werden bei der Anfangsaktualisierung hinzugefügt)

Diese fügen der Protokolltabelle Informationen hinzu, wenn ein Dokument in der Benutzertabelle hinzugefügt, gelöscht oder geändert wird. Die Informationen sind bei der nächsten geplanten oder manuellen Indexaktualisierung für die Indexsynchronisation erforderlich.

Beachten Sie, dass Trigger nur erstellt werden, wenn eine Protokolltabelle erstellt und der Textindex für eine Basistabelle und nicht für Sichten oder Kurznamentabellen erzeugt wird.

- Mit der erweiterten Protokoll- und Zwischenspeicherinfrastruktur:

Protokolltabelle

Sie verfolgt Aktualisierungen, die an den Dokumenten vorgenommen werden.

Eine Zusatztabelle für die Zwischenspeicherung

Sie verfolgt Einfügungen und Löschungen.

Ereignistabelle

Dient zum Sammeln von Informationen zu allen Aktualisierungen und möglichen Problemen während einer Aktualisierung der Textindizes.

Ein Aktualisierungstrigger für die Benutzertabelle (wird während der Anfangsaktualisierung hinzugefügt)

Der Aktualisierungstrigger fügt den Primärschlüssel der betreffenden Zeile zur Protokolltabelle hinzu, wenn ein Dokument in der indexierten Spalte aktualisiert wird.

Zur Optimierung der Leistung und der Nutzung von Plattenspeicherplatz verfügt der Befehl **CREATE INDEX** über eine Option zum Angeben eines anderen Tabellenbereichs für die Tabellen.

Anmerkung: Wenn Sie den Befehl **LOAD** zum Importieren Ihrer Dokumente verwenden, werden die Trigger nicht gestartet und eine inkrementelle Indexierung der geladenen Dokumente mit der regulären Infrastruktur ist nicht möglich. In diesem Fall ist die Verwendung des Befehls **DB2 IMPORT** vorzuziehen, da durch diesen Befehl die Trigger aktiviert werden.

Wenn die erweiterte textverwaltete Zwischenspeicherinfrastruktur für den Textindex konfiguriert ist, werden Dokumente, die mit der Operation **LOAD INSERT** eingefügt wurden, in der Zusatztabelle für die Zwischenspeicherung erfasst und die inkrementelle Indexierung ist möglich.

Beispiel

Im folgenden Beispiel wird ein Textindex für die Textspalte **HTMLFILE** der Tabelle **html tab** erstellt.

```
db2text create index DB2EXT.HTMLIDX for text on DB2EXT.HTMLTAB
      (HTMLFILE) format HTML
```

Für diese Tabelle muss ein Primärschlüssel vorhanden sein.

Die Standardwerte für die Indexerstellung werden der Sicht 'db2ext.dbdefaults' entnommen.

Zur Rücknahme der Änderungen, die durch den Befehl **CREATE INDEX** vorgenommen wurden, verwenden Sie den Befehl **DROP INDEX**. Informationen dazu finden Sie in Kapitel 37, „Löschen eines Textindex“, auf Seite 125.

Geben Sie den folgenden Befehl ein, um den erstellten Index mit Daten aus der Textspalte zu füllen:

```
db2text update index DB2EXT.HTMLIDX for text
```

Beachten Sie, dass Sie nach Dokumenten nur dann erfolgreich suchen können, wenn der Textindex mit der Tabelle unter Verwendung des Befehls **db2text update** synchronisiert ist.

Wenn bei der Indexierung Fehler auftreten, werden Zeilen für Indexaktualisierungseignisse zur Ereignistabelle hinzugefügt. Dazu kommt es, wenn beispielsweise ein Dokument nicht gefunden wird, das sich in der Warteschlange für die Indexierung befindet, oder wenn das Dokumentformat ungültig ist. Weitere Informationen finden Sie in der Beschreibung zur „Ereignissicht“ auf Seite 292.

Nächste Schritte

Anmerkung: Zusammenfassung der Suchmethoden

Abhängig von den bei der Indexerstellung ausgewählten Optionen sind verschiedene Suchmethoden möglich:

- Die skalaren SQL-Suchfunktionen arbeiten mit allen Textindizes mit Ausnahme solcher, die für Sichten erstellt wurden.
- Die Funktion zur Suche mit gespeicherten Prozeduren funktioniert nur für Textindizes, die mit einem Cache erstellt wurden.
- Die SQL-Tabellenwertfunktion funktioniert für alle Textindizes, einschließlich solcher, die für Sichten erstellt wurden.

Erstellen eines Textindex für Binärdatentypen

Wenn Sie Daten in einer Spalte mit einem Binärdatentyp (z. B. BLOB oder FOR BIT DATA) speichern, werden die Daten vom DB2-Datenbanksystem nicht konvertiert.

Informationen zu diesem Vorgang

Die Dokumente behalten ihre ursprünglichen Codepages (CCSIDs). Dies kann Probleme bei der Indexerstellung verursachen, da dann unter Umständen zwei verschiedene Codepages vorliegen. Daher müssen Sie festlegen, ob Sie die Codepage der Datenbank oder die im Befehl **CREATE INDEX** angegebene Codepage verwenden.

Um das Problem zu umgehen, geben Sie die Codepage beim Erstellen des Index an:

```
db2text CREATE INDEX db2ext.comment FOR TEXT ON db2ext.texttab (comment)
        CCSID 1252
```

Falls die Codepage nicht angegeben wurde, prüfen Sie, welche ID für codierten Zeichensatz (CCSID) zur Erstellung des Index verwendet wurde, indem Sie folgenden Befehl aufrufen:

```
db2 SELECT ccsid FROM db2ext.textindexes WHERE INDSHEMA = 'DB2EXT'
        and INDNAME = 'COMMENT'
```

Beachten Sie, dass die Verwendung von Dokumenten mit unterschiedlichen Codepages innerhalb eines Textindex nicht unterstützt wird. Informationen dazu, wie DB2-Produkte Einstellungen für Dokumentcodepages umwandeln, finden Sie im Handbuch Globalisierung.

Beachten Sie, dass sich das Problem nicht stellt, wenn Sie Indizes für Zeichendatentypen erstellen. Geben Sie bei Zeichendatentypen den Parameter **CCSID** nicht an.

Erstellen eines Textindex für einen nicht unterstützten Datentyp

Zur Erstellung eines Index müssen Textspalten bestimmte Datentypen aufweisen.

Informationen zu diesem Vorgang

Zur Erstellung eines Index müssen Textspalten einen der folgenden Datentypen aufweisen:

- CHAR
- VARCHAR
- LONG VARCHAR
- CLOB
- GRAPHIC
- VARGRAPHIC
- LONG VARGRAPHIC
- DBCLOB
- BLOB
- XML

Vorgehensweise

Wenn sich die Dokumente in einer Spalte mit einem nicht unterstützten Datentyp wie beispielsweise einem benutzerdefinierten Typ (UDT) befinden, müssen Sie Folgendes ausführen:

1. Stellen Sie eine Umsetzungsfunktion bereit, die den Benutzertyp als Eingabe verwendet und ihn in einen der gültigen Datentypen als Ausgabentyp umsetzt.
2. Geben Sie den Namen dieser Umsetzungsfunktion anschließend bei der Indexerstellung an. Weitere Informationen hierzu finden Sie in „Befehl CREATE INDEX“ auf Seite 227.

Beispiel

Sie wollen einen komprimierten Text in einer Tabelle speichern.

1. Erstellen Sie einen benutzerdefinierten Typ (UDT) für den Text in einer interaktiven SQL-Sitzung:

```
db2 "CREATE DISTINCT TYPE COMPRESSED_TEXT AS CLOB(1M)"
```
2. Erstellen Sie eine Tabelle, und fügen Sie den folgenden Text darin ein:

```
db2 "CREATE TABLE UDTTABLE (author VARCHAR(50) not null,  
                               text COMPRESSED_TEXT, primary key (author))"  
db2 "INSERT ..."
```
3. Erstellen Sie eine benutzerdefinierte Funktion (UDF), zum Beispiel mit dem Namen `uncompress`, die einen Wert des Typs `COMPRESSED_TEXT` empfängt und den entsprechenden dekomprimierten Text zum Beispiel als Wert des Typs `CLOB(10M)` zurückgibt.
4. Erstellen Sie Ihren Textindex auf folgende Weise, um die benutzerdefinierte Funktion `uncompress` anzugeben:

```
db2text "CREATE INDEX UDTINDEX for text ON UDTTABLE  
                                               (uncompress(text))  
                                               ..."
```

Erstellen eines Textindex für einen Kurznamen mit Indexteilaktualisierung unter Verwendung der DB2-Replikation

Bevor Sie einen Textindex für einen Kurznamen mithilfe einer Replikationserfassungstabelle erstellen, müssen Sie eine Reihe von Schritten ausführen.

Vorgehensweise

Bevor Sie einen Textindex für einen Kurznamen mithilfe einer Replikationserfassungstabelle erstellen, müssen Sie folgende Schritte ausführen:

1. Einrichten der föderierten DB2-Datenbank mit allen Serverdefinitionen und Wrapperdefinitionen.
2. Einrichten der Replikationssteuertabellen und Capture-Programme (Capture = Erfassung) auf dem fernen Server. Dort befindet sich die Quellentabelle für die Kurznamen. Wenn das DB2-Datenbanksystem nicht automatisch Kurznamen erstellt, müssen Sie diese in der föderierten DB2-Datenbank mithilfe eines Schemanamens für die folgenden Tabellen erstellen:

- IBMSNAP_SIGNAL
- IBMSNAP_PRUNE_SET
- IBMSNAP_PRUNCNTL
- IBMSNAP_REGISTER
- IBMSNAP_REG_SYNC (nur ferne Nicht-DB2-Datenbankserverquellen)

Nach Ausführung dieses Schrittes sind Kurznamen für die Replikationssteuertabellen in einem 'Capture-Steuerungsschema' in der föderierten DB2-Datenbank verfügbar. Dieser Schemaname ist für den Befehl **DB2TEXT CREATE INDEX** wichtig.

3. Registrieren Sie die Tabelle als Replikationsquelle.
4. Falls das DB2-Datenbanksystem im Arbeitsschritt für die Registrierung nicht automatisch einen Kurznamen erstellt, erstellen Sie einen Kurznamen für die Replikationserfassungstabelle in der föderierten Datenbank. Die Replikationserfassungstabelle kann entweder eine CD-Tabelle (CD = Change Data) oder eine CCD-Tabelle (CCD = Consistent Change Data) sein. Dieser Kurzname ist ein Parameter für den Befehl **DB2TEXT CREATE INDEX**.

Beachten Sie, dass die Spaltennamen IBMSNAP_OPERATION, IBMSNAP_COMMITSEQ und IBMSNAP_INTENTSEQ sowie die Namen der Primärschlüsselspalten nicht geändert werden dürfen.

5. Wenn Sie mit der DB2-Replikationsquelle arbeiten, müssen Sie sicherstellen, dass das Capture-Programm aktiv ist. Führen Sie für das Capture-Programm keinen Kaltstart aus: Wird ein Kaltstart ausgeführt, müssen sämtliche Zeilen in der Tabelle IBMSNAP_SIGNAL für APPLY_QUAL LIKE 'NSE%' erneut eingefügt werden. In der folgenden SQL-Anweisung ist die Vorgehensweise dargestellt:

```
INSERT INTO erfassungssteuerungsschema.IBMSNAP_SIGNAL
SELECT CURRENT TIMESTAMP, 'CMD', 'CAPSTART', MAP_ID, 'P'
FROM erfassungssteuerungsschema.IBMSNAP_PRUNCNTL
WHERE APPLY_QUAL LIKE 'NSE'
```

6. Sie können folgendes Beispiel für die Erstellung eines Textindex für einen Kurznamen durch Replikation verwenden:

```
DB2TEXT
CREATE INDEX indexname FOR TEXT ON kurzname (textspalte)
REPLICATION CAPTURE TABLE erfassungskurzname
CONTROL TABLE SCHEMA erfassungssteuerungsschema
```

Erstellen eines Textindex zur Verwendung durch die Suche mit gespeicherten Prozeduren

Beim Erstellen eines Textindex für die Suche mit gespeicherten Prozeduren müssen Sie bestimmte Parameter festlegen und berechnen, bevor Sie die Ausführung fortsetzen.

Informationen zu diesem Vorgang

Wenn Sie bereits vorab wissen, welche Datenuntergruppe Ihrer Tabelle Sie dem Benutzer bereitstellen möchten, und Sie nur die allerbesten Suchergebnisse und nicht die vollständige Ergebnisliste benötigen, können Sie die Suche mit gespeicherten Prozeduren verwenden. Um die Suche mit gespeicherten Prozeduren nutzen zu können, müssen Sie bei der Ausführung des Befehls **CREATE INDEX** Cacheoptionen angeben. Das Arbeiten mit einem Cachindex ermöglicht bei der Abfrage durch Verschieben aller angegebenen Daten in den Hauptspeicher eine hohe Leistung; dadurch werden kostenintensive physische Leseoperationen aus der Tabelle vermieden.

Bevor Sie den Cachindex zum ersten Mal aktualisieren, stellen Sie sicher, dass sich in Ihrer Tabelle bereits Dokumente befinden; dadurch wird die Aktualisierung eines Index für eine nicht befüllte Tabelle vermieden. Damit werden eine bessere Indexierungsleistung und eine korrekte Einschätzung der Anforderungen des Caches gewährleistet.

Die Suche mit gespeicherten Prozeduren ermöglicht Ihnen, vordefinierte Daten schnell abzurufen, die einem Dokument zugeordnet sind. Um dies im Befehl **CREATE INDEX** zu definieren, verwenden Sie die Option für Cachetabellen. Mit dem Befehl **ACTIVATE CACHE** werden die angegebenen Daten anschließend in den Hauptspeichercache verschoben.

Beim Erstellen eines Textindex für die Suche mit gespeicherten Prozeduren müssen Sie folgende Parameter festlegen und berechnen:

- Typ des Caches (temporär oder persistent).
- Type der Indexaktualisierung (automatisch und inkrementell bzw. Neuerstellung bei jeder Aktualisierung)
- Maximale Größe des Speichers, der von Net Search Extender unter Verwendung von `MAXIMUM_CACHE_SIZE` genutzt werden kann.
- Größe des freien Speicherbereichs, der für nachfolgende Dokumentaktualisierungen erforderlich ist (mithilfe von `PCTFREE`). Beachten Sie, dass dies nur für Teilaktualisierungen gilt.

Folgende Cachetypen stehen zur Verfügung:

Temporärer Cache

Dieser wird für jeden Befehl **DB2TEXT ACTIVATE CACHE** neu erstellt; außerdem ist ein erneutes Laden der Daten aus Ihrer DB2-Tabelle in den Speicher erforderlich. Die komplette Neuerstellung des Cache-Index bei jedem Neustart von Net Search Extender bzw. bei jedem Systemneustart nimmt mehr Zeit in Anspruch als die Reaktivierung eines persistenten Caches, insbesondere bei großen Tabellen. Verwenden Sie einen temporären Cache nur, wenn Sie mit einer kleinen Menge an Festdaten arbeiten und keine Rücksicht auf die Zeitdauer für die Erstellung von im Cache gespeicherten Daten nehmen müssen.

Persistenter Cache

Dieser wird auf der Platte belassen und kann mithilfe des Befehls **DB2TEXT ACTIVATE CACHE** rasch dem Speicher zugeordnet werden. In Szenarios mit Indexteilaktualisierungen muss der Cache aktiviert bleiben, um eine Synchronisierung zwischen der Tabelle und dem in den Cache gestellten Index zu ermöglichen. Anderenfalls erstellt der nächste Befehl **DB2TEXT ACTIVATE CACHE** den Cache völlig neu.

Folgende Methoden für die Aktualisierung eines Textindex sind verfügbar:

Ohne Option **RECREATE INDEX ON UPDATE**

Wenn die Option **RECREATE INDEX ON UPDATE** nicht gesetzt ist, findet eine automatische Indexaktualisierung statt. Der Prozess wird durch den Befehl zur Indexaktualisierung ausgelöst, und die Aktualisierungsintervalle werden durch die Option für die Aktualisierungsfrequenz festgelegt. Der Aktualisierungsprozess wird auch Teilaktualisierung genannt.

Vermeiden Sie das Löschen und erneute Einfügen eines Dokuments in die Tabelle, da Slots für ein gelöscht Dokument im Cache nicht wiederverwendet werden können. Sie sollten folglich vermeiden, Spalten eines aktivierten Index zu ändern.

Mit Option **RECREATE INDEX ON UPDATE**

Damit wird der Index bei jeder Aktualisierung neu erstellt. Verwenden Sie nach Möglichkeit in den Cachespaltenausdrücken *variable*-Datentypen. Dadurch wird Cachespeicher gespart. Verwenden Sie entsprechende Umsetzungsausdrücke (CAST) in der Klausel **CACHE TABLE**.

Verwenden Sie diese Option, wenn Ihre Daten nicht allzu stabil sind, d. h., wenn Sie erwarten, dass mehr als 50% Ihrer Dokumente nach der ersten Indexaktivierung eingefügt werden.

Net Search Extender stellt zwei SQL-Funktionen bereit, die Sie bei der Bestimmung der Speicherparameter für den Befehl **CREATE INDEX** unterstützen: **MAXIMUM_CACHE_SIZE** und **PCTFREE**.

- **MAXIMUM_CACHE_SIZE** gibt die maximale Größe des im Cache gespeicherten Index an. Mithilfe der folgenden benutzerdefinierten Funktion können Sie den Wert von **MAXIMUM_CACHE_SIZE** in Megabyte (MB) abrufen:

```
DB2EXT.MAXIMUM_CACHE_SIZE(maximumNumberDocs INTEGER,  
                           averageRowLength INTEGER, numberOfCacheColumns INTEGER)
```

Folgender Befehl gibt den Parameter für die durchschnittliche Zeilenlänge Ihrer Tabelle zurück:

```
SELECT AVG(LENGTH(cachespalte_1) + ... + LENGTH(cachespalte_n))
```

Beachten Sie, dass sich dieser Durchschnittswert wesentlich ändern kann, wenn weitere Werte in die Tabelle eingefügt werden. Die Anzahl der Cachespalten bezieht sich auf die Anzahl von Spaltenausdrücken, die Sie in der Klausel **CACHE TABLE** des Befehls **DB2TEXT CREATE INDEX** angegeben haben.

Weitere Informationen finden Sie in Kapitel 15, „Suche mit gespeicherten Prozeduren - Speicherbedarf“, auf Seite 63.

- **PCTFREE** gibt an, welcher Prozentsatz des in **MAXIMUM_CACHE_SIZE** angegebenen Caches für weitere Dokumente freigehalten werden soll. Die folgende benutzerdefinierte Funktion gibt den empfohlenen Wert für **PCTFREE** auf der Grundlage der tatsächlichen und der maximalen Anzahl von Dokumenten zurück.

```
DB2EXT.PCTFREE(actualNumberDocs INTEGER, maximumNumberDocs INTEGER)
```

Die tatsächliche Anzahl von Dokumenten ist die Anzahl von Zeilen in Ihrer Tabelle zum Zeitpunkt der ersten Ausführung des Befehls **ACTIVATE CACHE**, wodurch der Speichercache erstellt wird.

Die maximale Anzahl von Dokumenten ist ein Schätzwert für die maximale Anzahl von Dokumenten in Ihrer Tabelle vor der nächsten Ausführung des Befehls **DB2TEXT ACTIVATE** (für einen temporären Cache) bzw. des Befehls **DB2TEXT ACTIVATE CACHE RECREATE** (für einen persistenten Cache).

Der Standardwert ist 50%. Falls Sie den Index bei jeder Aktualisierung neu erstellen, setzen Sie den Wert **PCTFREE** auf 0.

Beispiel

Nehmen Sie an, Ihre Tabelle hat 10.000 Zeilen, und Sie erwarten insgesamt maximal 20.000 Zeilen. Verwenden Sie folgenden Aufruf zur Berechnung des benötigten Wertes für **PCTFREE**:

```
db2 "values DB2EXT.PCTFREE(10000,20000) "
```

Angenommen, die maximale Zeilenzahl beträgt 20.000, und in Ihrem Cache befinden sich zwei Spalten mit einer durchschnittlichen Größe von 76. Verwenden Sie folgenden Aufruf, um die Größe zurückzugeben:

```
db2 " values DB2EXT.MAXIMUM_CACHE_SIZE(20000,76,2) "
```

Nächste Schritte

Nach der Ermittlung geeigneter Parameterwerte können Sie Ihren im Cache gespeicherten Index mithilfe des folgenden Aufrufs erstellen:

```
db2text CREATE INDEX db2ext.comment FOR TEXT ON db2ext.texttab (comment)
          CACHE TABLE (docid) PCTFREE 10 MAXIMUM CACHE SIZE 5
```

In diesem Beispiel wird die Spalte docid im Cache gespeichert, wobei der Hauptspeicher zur raschen Rückgabe einer Ergebnistabelle verwendet wird. Zehn Prozent des Cachespeichers werden für zukünftige Dokumente reserviert. Der Cache wird auf eine Maximalgröße von 5 MB begrenzt.

Kapitel 29. Erstellen von Textindizes für Sichten

Mit den Suchfunktionen für gespeicherte Prozeduren bzw. für Tabellenwerte können Sie Textindizes für Sichten erstellen.

Informationen zu diesem Vorgang

Sie können jedoch keine der Skalarfunktionen (z. B. CONTAINS) verwenden. Eine weitere gravierende Einschränkung besteht darin, dass für Sichten keine Trigger erstellt werden können. Somit werden in den zugrunde liegenden Basistabellen Änderungen nicht automatisch erkannt.

Bei Indexteilaktualisierungen muss der Benutzer daher wissen, welches Dokument hinzugefügt, aktualisiert oder gelöscht wurde, um den Textindex mit der Datenbank zu synchronisieren. Zu diesem Zweck müssen alle Änderungen der Protokolltabelle hinzugefügt werden. Dieser Prozess wird im folgenden Beispiel illustriert:

Beispiel

1. Verwenden Sie den folgenden Befehl, um eine Basistabelle zu erstellen:

```
db2 "create table DB2EXT.TLOGIX140789
(key INTEGER not null PRIMARY KEY,
name VARCHAR(50) not null, comment VARCHAR(90))"
```

2. Verwenden Sie die folgenden Befehle, um einige Einträge hinzuzufügen:

```
db2 "insert into DB2EXT.TLOGIX140789 values
(1,'Claus','works in room 301')"
db2 "insert into DB2EXT.TLOGIX140789 values
(2,'Manja','is in the same office as Juergen')"
db2 "insert into DB2EXT.TLOGIX140789 values
(2,'Juergen','has the longest way to Raiko')"
db2 "insert into DB2EXT.TLOGIX140789 values
(3,'Raiko','is sitting in the office besides Claus ')"
```

3. Verwenden Sie den folgenden Befehl, um eine Sicht zu erstellen:

```
db2 "create view sampleview as select key, comment from DB2EXT.TLOGIX140789"
```

4. Verwenden Sie die folgenden Befehle zum Erstellen, Aktualisieren und Aktivieren des Textindex:

```
db2text "create index indexview for text on sampleview(comment)
        cache table (comment) maximum cache size 1 key columns
        for index on view (key)"
db2text "update index indexview for text"
db2text "activate cache for index indexview for text"
```

Anmerkung: Sie müssen die Cachetabelle angeben, um einen Textindex für eine Sicht erstellen zu können. Zur Erstellung der korrekten Protokolltabelle müssen Sie die Spalten für den Index in einer Sicht angeben. Wenn Sie auf diese Weise einen Index erstellen, können Sie den Index auch mit der Tabellenwertfunktion durchsuchen.

Wenn Sie die Suche mit gespeicherten Prozeduren in einer Umgebung mit partitionierten Datenbanken verwenden, müssen Sie für Verwaltungstabellen einen Tabellenbereich in einer einzigen Partition explizit angeben und diese Partition explizit aufrufen. Um sicherzustellen, dass Sie eine Verbindung zur richtigen Partition herstellen, verwenden Sie die Umgebungsvariable **DB2NODE**.

5. Verwenden Sie zur Aktualisierung der Tabelle die folgenden Befehle:


```
db2 "insert into DB2EXT.TLOGIX140789 values
(4,'Bernhard','is working on the same floor
      as Manja, but not as Claus')"
db2 "insert into DB2EXT.TLOGIX140789 values
(5,'Guenter','shares the office with Raiko')"
```
6. Aktualisieren Sie anschließend die Protokolltabelle. Geben Sie folgenden Befehl ein, um den Namen der Protokolltabelle abzurufen:


```
db2 "select INDSHEMA,INDNAME,LOGVIEWSCHEMA,LOGVIEWNAME
      from db2ext.textindexes"
```

Das Layout der Protokolltabelle sieht folgendermaßen aus:

sqltype	sqllen	sqlname.data	sqlname.length
496	INTEGER	4 OPERATION	9
392	TIMESTAMP	26 TIME	4
497	INTEGER	4 PK01	4

Verwenden Sie folgende Befehle, um die Einträge der Protokolltabelle hinzuzufügen:

```
db2 "insert into DB2EXT.TLOGIX140789 values(0,CURRENT TIMESTAMP,4)"
db2 "insert into DB2EXT.TLOGIX140789 values(0,CURRENT TIMESTAMP,5)"
```

Der erste Wert beschreibt die Operation (0 = Einfügen, 1 = Aktualisieren, 2 = Löschen). Der zweite muss immer CURRENT TIMESTAMP lauten, und der letzte Wert ist der Primärschlüssel der Zeile, die eingefügt, aktualisiert oder gelöscht wurde.

7. Verwenden Sie folgenden Befehl, um den Index erneut zu aktualisieren:


```
db2text "update index indexview for text"
```

Sie können nun mithilfe der gespeicherten Prozedur nach den neuen Werten suchen.

Kapitel 30. Erstellen eines Textindex für bereichspartitionierte Tabellen

Sie können Textindizes für bereichspartitionierte Tabellen mit oder ohne die erweiterte textverwaltete Zwischenspeicherinfrastruktur erstellen, die inkrementelle Indexaktualisierungen unterstützt.

Informationen zu diesem Vorgang

Zum Inaktivieren der Infrastruktur für einen Textindex in einer bereichspartitionierten Tabelle geben Sie den Befehl **CREATE INDEX** mit dem auf **OFF** gesetzten Parameter **AUXLOG** an, wie im folgenden Beispiel gezeigt:

```
db2text create index sampleix for text on sample(comment) administration tables
in mytablespace index configuration(auxlog off) connect to mydb
```

In diesem Fall wird die primäre Protokolltabelle hinzugefügt und Dokumentänderungen werden über Trigger erkannt. Beachten Sie, dass die Klausel **ADMINISTRATION TABLES IN** verwendet werden muss, wenn Indizes für bereichspartitionierte Tabellen erstellt werden, da andernfalls ein Fehler auftritt.

Sie können eine Teilaktualisierung nicht verwenden, um Änderungen zu verarbeiten, die sich auf zugeordnete Bereiche bzw. auf Bereiche beziehen, für die die Zuordnung aufgehoben ist, oder um Dokumente zu verarbeiten, die Sie mit dem Befehl **LOAD** in Verbindung mit dem Parameter **INSERT** in eine hinzugefügte Partition geladen haben. Sie müssen den Textindex erneut erstellen, um ihn mit der Basistabelle zu synchronisieren.

Wenn die erweiterte textverwaltete Zwischenspeicherinfrastruktur für den Textindex aktiviert ist, werden Dokumentaktualisierungen über einen Aktualisierungstrigger in der primären Protokolltabelle erfasst, während Dokumenteinfügungen und -löschungen über die Integrationsverarbeitung in der Zusatztabelle für die Zwischenspeicherung erfasst werden. Dieser Prozess wird in den folgenden Beispielszenarios gezeigt:

Beispiel

Szenario 1: Mit der erweiterten textverwalteten Zwischenspeicherinfrastruktur Partition für eine Tabelle zuordnen

```
db2 "create table uc_007_customer_archive (pk integer not null primary key,
customer varchar(128) not null, year integer not null,
address blob(1M) not null) partition by range(year)(starting(2000)ending(2001)every 1)"
db2text "create index uc_007_idx for text on uc_007_customer_archive (address)
administration tables in mytablespace"
db2 "select indexname, logviewname, auxstagingname from db2ext.textindexes"
db2text "update index uc_007_idx for text"
db2 "create table uc_007_customer_2001 (pk integer not null primary key,
customer varchar(128) not null, year integer not null, address blob(1M) not null)"
db2 "import from uc_007_2001.del of del lobes from ./data modified by codepage=1208
insert into uc_007_customer_2001"
db2 "alter table uc_007_customer_archive attach partition p2001 starting(2001) ending(2002)
exclusive from uc_007_customer_2001"
```

Beachten Sie, dass die Änderungen bisher nicht sichtbar sind und dass eine Integritätsverarbeitung erforderlich ist.

```
db2 "select * from sysibmts.sysauxlog_ix253720"
PK      GLOBALTRANSID      GLOBALTRANSTIME      OPERATIONTYPE
-----
0 record(s) selected.
```

```
db2 "set integrity for uc_007_customer_archive immediate checked"
```

Mit der Integritätsverarbeitung werden abhängige Tabellen in den Modus 'Anstehend' versetzt.

```
db2 "select * from sysibmts.sysauxlog_ix253720"
PK      GLOBALTRANSID      GLOBALTRANSTIME      OPERATIONTYPE
-----
SQL0668N Die Operation ist wegen Ursachencode '1' für Tabelle
'SYSIBMTS'.'SYSTAUXLOG_IX253720' nicht zulässig. SQLSTATE=57016
```

Führen Sie die Integritätsverarbeitung für die textverwaltete(n) Zwischenspeichertabelle(n) aus. Mit dem Befehl werden alle Textindizes für die Tabelle verarbeitet.

```
db2text "reset pending for table uc_007_customer_archive for text"
```

```
db2 "select * from sysibmts.sysauxlog_ix253720"
PK      GLOBALTRANSID      GLOBALTRANSTIME      OPERATIONTYPE
-----
1  x'000000000002215B'    x'20081020204612500381000000'    1
2  x'000000000002215B'    x'20081020204612500602000000'    1
3  x'000000000002215B'    x'20081020204612500734000000'    1
5  x'000000000002215B'    x'20081020204612500864000000'    1
```

Die Teilaktualisierung verarbeitet die Daten aus der neu zugeordneten Partition.

```
db2text "update index uc_007_idx for text"
```

Szenario 2: Mit der erweiterten textverwalteten Zwischenspeicherinfrastruktur Zuordnung einer Partition für eine Tabelle aufheben

```
db2 alter table uc_007_customer_archive detach partition p2005 into t4p2005
SQL3601W
```

Die Anweisung führte dazu, dass eine oder mehrere Tabellen automatisch in den Status 'Festlegen der Integrität anstehend' versetzt wurden.
SQLSTATE=01586

```
db2text "reset pending for table uc_007_customer_archive for text"
```

```
db2text "update index uc_007_idx for text"
```

Kapitel 31. Leistungsaspekte für die Indexierung

Bei der Verbesserung der Indexierungsleistung muss eine Reihe von Aspekten beachtet werden.

Im Hinblick auf die Verbesserung der Leistung beim Indexieren sollten Sie die folgenden Gesichtspunkte beachten:

- Verwenden Sie zum Speichern der Textdokumente den Datentyp VARCHAR und nicht LONG VARCHAR oder CLOB.
- Verwenden Sie zum Speichern des Textindex und der Datenbankdateien verschiedene physische Platten.
- Verwenden Sie anstelle von VARCHAR-Typen kleine Primärschlüsselspalten, beispielsweise TIMESTAMP und INTEGER.
- Stellen Sie sicher, dass Ihr System über ausreichend Realspeicher für alle diese Daten verfügt. Wenn nicht genügend Speicher zur Verfügung steht, arbeitet das Betriebssystem stattdessen mit der Auslagerung von Speicher. Dadurch wird die Suchleistung und das Indexieren vermindert.
- Der Aktualisierungsparameter `commitcount`, der bei der automatischen oder manuellen Aktualisierung des Index verwendet wird, verlangsamt bei der Teilindexierung die Leistung der Indexierung. Beachten Sie, dass der Parameter nicht während des Anfangsaktualisierungsprozesses verwendet wird.
- Wenn viele Fehlernachrichten und Warnungen in die Tabelle des Ereignisprotokolls geschrieben werden, kann es während der Indexaktualisierung zu Leistungsbeeinträchtigungen kommen.

Teil 10. Pflegen von Textindizes

Für die Pflege von Textindizes müssen einige Verwaltungstasks ausgeführt und aussagekräftige Informationen zum Indexstatus abgerufen werden.

In diesem Abschnitt wird die Pflege von Textindizes sowie das Abrufen nützlicher Statusinformationen beschrieben. Zur Pflege von Indizes gehören die folgenden Aufgaben:

1. Aktualisieren und Reorganisieren eines Textindex.
2. Ändern eines Textindex.
3. Bereinigen (Löschen) von Informationen zum Indexaktualisierungsereignis.
4. Löschen eines Textindex.
5. Anzeigen von Indexstatus.

Dieser Abschnitt enthält darüber hinaus Informationen zum Backup und Restore von Indizes und aktivierten Datenbanken.

Kapitel 32. Aktualisieren und Reorganisieren eines Textindex

Nachdem Sie den Textindex zum ersten Mal erstellt und aktualisiert haben, muss der Index stets auf dem neuesten Stand gehalten werden. Wenn Sie beispielsweise ein Textdokument zu einer Tabelle hinzufügen oder ein vorhandenes Dokument in einer Tabelle ändern, muss das Dokument indexiert werden, um den Inhalt des Index mit dem Inhalt der Tabelle synchron zu halten. Ebenso müssen beim Löschen eines Textdokuments aus einer Tabelle die zugehörigen Referenzen aus dem Index entfernt werden.

Geben Sie die Option **RECREATE** im Befehl **CREATE INDEX** an, wird der Index bei jeder Aktualisierung vollständig neu erstellt. Mit dieser Option werden keine Protokolltabellen oder Trigger erstellt. Wenn Sie über umfangreiche Tabellen verfügen, sollten Sie diese Option mit Sorgfalt anwenden, da der Rebuild des vollständigen Index kostenintensiv sein kann.

Wurde der Textindex ohne die Option **RECREATE INDEX ON UPDATE** erstellt, werden Informationen zu neuen, geänderten oder gelöschten Dokumenten über Trigger in einer Protokolltabelle gespeichert. Wenn der Textindex mit der erweiterten textverwalteten Zwischenspeicherinfrastruktur (**AUXLOG ON**) konfiguriert wurde, speichert ein Aktualisierungstrigger Informationen zu geänderten Dokumenten in der Protokolltabelle, während Informationen zu Einfügungen und Löschungen über die Integrationsverarbeitung in der Zusatztabelle für die Zwischenspeicherung gespeichert werden.

In der Regel aktualisieren Sie einen Index automatisch in bestimmten Zeitabschnitten. Die Aktualisierungsfrequenz kann für einen vorhandenen Index mithilfe des Befehls **ALTER INDEX** geändert werden.

Die Indexaktualisierungsfrequenz wird in Bezug auf den Zeitpunkt, zu dem die Aktualisierung auszuführen ist, und die Mindestanzahl von Textänderungen angegeben, die sich in der Protokolltabelle vor der Indexaktualisierung in der Warteschlange befinden müssen. Wenn zum angegebenen Zeitpunkt (Tag und Uhrzeit) nicht genügend Änderungen in der Protokolltabelle aufgezeichnet sind, wird der Index nicht aktualisiert.

Sie sollten die regelmäßige Indexierung sorgfältig planen, da das Indexieren großer Mengen von Textdokumenten ein zeitaufwendiger und ressourcenintensiver Prozess sein kann. Die erforderliche Zeit ist von vielen Faktoren abhängig. Hierzu gehört z. B. die Größe der Dokumente, die Anzahl der Textdokumente, die seit der vorigen Indexaktualisierung hinzugefügt oder geändert wurden, und die Leistungskapazität des Prozessors. Hier zwei wichtige Tipps für die Planung von Indexaktualisierungen:

- Beim Arbeiten in einer Umgebung mit partitionierten Datenbanken müssen Sie besonderes Augenmerk auf den Plan für die Aktualisierung des Index legen, damit die Anzahl von gleichzeitig ablaufenden Indexaktualisierungen minimiert wird. Dies ist deshalb wichtig, weil bei geplanten Aktualisierungen von Textindizes im System das gleichzeitige Ablaufen mehrerer Indexaktualisierungen dazu führen kann, dass so viele Prozesse pro Partition vorhanden sind, wie Indizes zu einem bestimmten Zeitpunkt aktualisiert werden.

- Sie sollten die Kombination einer großen Anzahl von Indizes und sehr hoher automatischer Aktualisierungsfrequenzen vermeiden, da dies zu Sperren führen kann. Beispiel: 100 Indizes mit einer Aktualisierungsfrequenz von 5 Minuten, 24 Stunden am Tag und 7 Tagen die Woche generieren eine interne Liste mit $100 \cdot 12 \cdot 24 \cdot 7 = 201600$ Prüfpunkten pro Woche, die verwaltet werden müssen.

Anmerkung: In folgenden Fällen können für eine DB2-Tabelle Rollbacks oder Deadlocks auftreten.

- Hohe Aktualisierungshäufigkeit
- Hohe Häufigkeit von Änderungstransaktionen
- Lang andauernde Transaktionen

Wenn eine Datenbanktabelle aktualisiert wird, müssen die für den Net Search Extender-Index erforderlichen Änderungen in einer Protokolltabelle protokolliert werden. Wenn diese Protokolltabelleneinträge verarbeitet wurden, werden die Einträge aus der Protokolltabelle gelöscht. Wenn sich diese Löschoperationen für die Protokolltabelle mit den Aktualisierungen der Datenbanktabelle, die protokolliert werden müssen, überschneiden, kann es zu einem Deadlock kommen.

Wenn die erweiterte textverwaltete Zwischenspeicherinfrastruktur für den Index konfiguriert ist, blockieren bestimmte Datenbankoperationen für die Basistabelle möglicherweise den Zugriff auf die Zusatztabelle für die Zwischenspeicherung. Stellen Sie vor dem Aktualisieren des Textindex sicher, dass sich die Zusatztabelle für die Zwischenspeicherung nicht im Modus 'Anstehend' befindet.

Kapitel 33. Aktualisieren eines Textindex

Der Befehl **UPDATE INDEX** ermöglicht Ihnen, einen Index unverzüglich auf Anforderung zu aktualisieren.

Informationen zu diesem Vorgang

Wann Wenn ein Index sofort aktualisiert werden muss, ohne auf die festgelegte regelmäßige Indexierung zu warten.

Befehl UPDATE INDEX

Berechtigung

Zu den Zugriffsrechten, über die die Berechtigungs-ID der Anweisung verfügt, muss mindestens eines der folgenden gehören:

- Das Zugriffsrecht CONTROL für die Tabelle, für die der Index definiert ist.
- Das Zugriffsrecht DATAACCESS.

Mit folgendem Befehl wird der Index aktualisiert:

```
db2text UPDATE INDEX comment FOR TEXT
```

Dieser Befehl ist nützlich, wenn Sie einer Datenbank mehrere Textdokumente hinzugefügt haben und diese unverzüglich durchsuchen wollen.

Wenn Sie **AUTOMATIC REORGANIZE** im Befehl **CREATE INDEX** angeben, wird der Index automatisch reorganisiert, wenn dies erforderlich ist.

Wenn Sie stattdessen **MANUAL REORGANIZATION** angeben und feststellen möchten, ob eine manuelle Reorganisation erforderlich ist, fragen Sie die Sicht 'db2ext.textindexes' mit folgendem Befehl ab:

```
db2 "select reorg_suggested from db2ext.textindexes where INDNAME = 'comment'"
```

Wenn Sie **MANUAL REORGANIZATION** angeben und häufig Aktualisierungen für eine Spalte ausführen, beachten Sie, dass sich der Aktualisierungsprozess verlangsamt. Geben Sie zur manuellen Reorganisation den folgenden Befehl ein:

```
db2text UPDATE INDEX comment FOR TEXT reorganize
```

Kapitel 34. Ändern eines Textindex

Wenn die Aktualisierungshäufigkeit bzw. das Index- oder das Arbeitsverzeichnis geändert werden müssen, setzen Sie hierzu den Befehl **ALTER INDEX** ab.

Vorbereitende Schritte

Zu den Zugriffsrechten, über die die Berechtigungs-ID der Anweisung verfügt, muss mindestens eines der folgenden gehören:

- Das Zugriffsrecht CONTROL für die Tabelle, für die der Index definiert ist.
- Die Berechtigung DBADM.

Informationen zu diesem Vorgang

Mit dem Befehl **ALTER INDEX** können Sie das Index- und Arbeitsverzeichnis, die Aktualisierungshäufigkeit eines Index oder die Cachemerkmale wie z. B. die Werte für **MAXIMUM CACHE SIZE** oder **PCTFREE** ändern. Wenn Sie keine Aktualisierungshäufigkeit angeben, bleiben die aktuellen Einstellungen unverändert. Wird gerade eine Indexaktualisierung oder eine Suche ausgeführt, erscheint eine Fehlermeldung. Diese gibt an, dass der Index zurzeit gesperrt ist und keine Änderungen ausgeführt werden können.

Beispiel

Im folgenden Beispiel wird die Aktualisierungshäufigkeit für den Index geändert.

```
db2text ALTER INDEX comment FOR TEXT
        UPDATE FREQUENCY d(1,2,3,4,5) h(12,15) m(00) UPDATE MINIMUM 100
```

In diesem Beispiel wird der Index um 12:00 oder um 15:00 von Montag bis Freitag aktualisiert, wenn sich mindestens 100 Dokumente in der Warteschlange befinden.

Verwenden Sie folgenden Befehl, um die regelmäßige Aktualisierung eines Index zu stoppen:

```
db2text ALTER INDEX comment FOR TEXT
        UPDATE FREQUENCY NONE
```

Wenn Sie die Indexverzeichnisse mit dem Befehl **ALTER INDEX** ändern, werden die Indexdateien aus dem ursprünglichen Indexverzeichnis an eine neue Speicherposition versetzt und der Index wird während dieses Prozess gesperrt. Bei großen Indizes und bei Änderungen über Dateisysteme hinweg kann dies eine beträchtliche Menge an Zeit in Anspruch nehmen. Nach dem Abschluss des Kopiervorgangs wird die Sperre wieder aufgehoben, sodass der Index wieder verwendet werden kann.

Kapitel 35. Löschen von Indexereignissen

Wenn Sie die Nachrichten in der Ereignistabelle eines Index nicht mehr benötigen, setzen Sie den Befehl **CLEAR EVENTS FOR INDEX** ab.

Vorbereitende Schritte

Die Berechtigungs-ID der Anweisung muss mindestens eine der folgenden Berechtigungen umfassen:

- Das Zugriffsrecht CONTROL für die Tabelle, für die der Index definiert ist.
- Die Berechtigung DBADM.

Informationen zu diesem Vorgang

In der Ereignistabelle des Index werden Informationen über Indexierungsereignisse gespeichert, wie zum Beispiel die Zeitpunkte für den Start und das Ende von Aktualisierungen, die Anzahl indexierter Dokumente oder Dokumentfehler, die bei der Aktualisierung aufgetreten sind. Die Ereignistabelle kann bei der Ermittlung der Ursache des Problems hilfreich sein. Wenn Sie diese Nachrichten nicht mehr benötigen, können Sie sie löschen.

Beispiel

Im folgenden Beispiel werden Nachrichten aus dem angegebenen Textindex gelöscht:

```
db2text CLEAR EVENTS FOR INDEX comment FOR TEXT
```

Kapitel 36. Prüfen eines Textindex (Windows, AIX)

Ab DB2 Version 10 Fixpack 1 können Sie das Net Search Extender-Dienstprogramm für die Indexprüfung (Befehl **checknseindex**) dazu verwenden, sicherzustellen, dass ein Textindex korrekt funktioniert.

Vorbereitende Schritte

Stellen Sie sicher, dass für den Textindex, den Sie prüfen möchten, keine Aktualisierungs- oder Löschoptionen in Bearbeitung sind.

Informationen zu diesem Vorgang

Textindizes müssen auf mögliche Beschädigung hin überprüft werden, wenn eines oder mehrere der nachfolgend aufgeführten Szenarios auftreten:

- Suchabfragen geben Fehler zurück, jedoch ohne bestimmte Fehlercodes oder Erläuterungen.
- In den Suchergebnissen sind keine neu hinzugefügten oder aktualisierten Textdokumente berücksichtigt.
- Eine Indexaktualisierung schlägt fehl.
- Das Ereignisprotokoll enthält Kernelfehlernachrichten.

Vorgehensweise

Geben Sie zur Prüfung eines Textindex den Befehl **checknseindex** ein und verwenden Sie dabei zumindest die Parameter **-i** und **-p**. Dieses Dienstprogramm benötigt eine erhebliche Menge an Systemressourcen; abhängig von der Größe des Textindex kann die Ausführung des Dienstprogramms viel Zeit in Anspruch nehmen.

Ergebnisse

Beispiel 1: Verwenden Sie den Befehl **checknseindex**, um den Status eines gültigen Textindex zu prüfen.

```
C:\SQLLIB\bin\checknseindex.exe -p "C:\litu_ict\Corrupted Index\NODE0000"  
-i IX335811
```

```
CTE5265I Das NSE-Indexprüfungsprogramm hat festgestellt, dass der angegebene Index,  
"IX335811", gültig ist.
```

Keine Aktion erforderlich.

Beispiel 2: Verwenden Sie den Befehl **checknseindex**, um den Status eines gültigen Textindex zu prüfen, dessen Pfad ungültig ist:

```
C:\SQLLIB\bin\checknseindex.exe -p "C:\litu_ict\Corrupted Index\NODE00001"  
-i IX335811
```

```
CTE5254E Das NSE-Indexprüfungsprogramm konnte den angegebenen Index nicht prüfen,  
weil das Dienstprogramm nicht auf den Pfad "C:\litu_ict\Corrupted Index\NODE00001"  
zugreifen konnte.
```

Korrigieren Sie den Indexpfad und geben Sie den Befehl erneut ein.

Beispiel 3: Verwenden Sie den Befehl **checknseindex** mit den Parametern **-deepCheck** und **-v**, um den Status eines ungültigen Textindex zu prüfen:
C:\SQLLIB\bin\checknseindex.exe -p "C:\litu_ict\Corrupted Index\NODE0000"
-i IX335812 -deepCheck -v

```
=====
checknseindex
-----

Net Search Extender-Indexprüfungsprogramm

=====

NSE-Zuordnungsindizes werden geprüft...

    Datensteuersatz für progressive Zuordnung: Anzahl Blöcke = "1"
    Datensteuersatz für regressive Zuordnung: Anzahl Blöcke = "1"
    Indexsteuersatz für progressive Zuordnung: Anzahl Dokumente = "8"
    Indexsteuersatz für regressive Zuordnung: Anzahl Dokumente = "8"

Interne NSE-Indizes werden geprüft...

Fri Mar 30 13:39:40 2012 Primärindex prüfen...
    100%
Fri Mar 30 13:39:40 2012 Sekundärindex prüfen...
    100%
Fri Mar 30 13:39:40 2012 Ende

Konsistenz der NSE-Zuordnungsindizes und der internen NSE-Indizes wird geprüft...

CTE5263E Das NSE-Indexprüfungsprogramm hat festgestellt, dass der
NSE-Zuordnungsindex und der interne NSE-Index nicht konsistent sind.
Ursachencode: "1". Diagnosedaten: "".
```

Versuchen Sie, den beschädigten Index zu korrigieren.

Nächste Schritte

- Wenn ein Index beschädigt ist und nicht mehrere Gigabyte an Festplattenspeicherplatz belegt, erstellen Sie ihn neu. Weitere Informationen hierzu finden Sie im Abschnitt zur Erstellung eines Textindex.
- Wenn der ungültige Index mehrere Gigabyte an Festplattenspeicherplatz belegt, wenden Sie sich an die IBM Kundenunterstützung.
- Wenn Sie für die Net Search Extender-Textindizes ein Backup durchgeführt haben und ein Dokumentverwaltungssystem verwenden, führen Sie die entsprechenden Schritte der Net Search Extender-Prozedur für die Recovery von Textindizes aus.

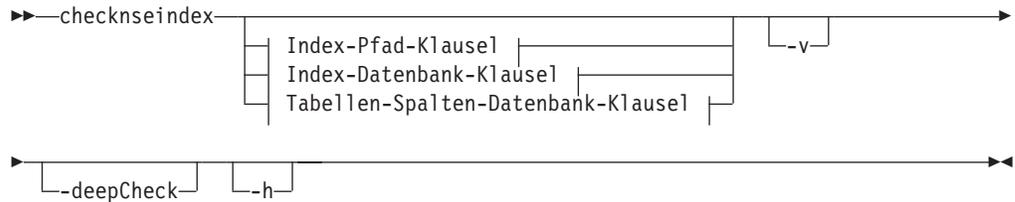
Befehl 'checknseindex' (Windows, AIX)

Ab DB2 Version 10.1 Fixpack 1 können mit diesem Befehl Beschädigungen oder andere Abweichungen in Net Search Extender-Textindizes festgestellt werden. Mit diesem Befehl kann die Konsistenz zwischen Zuordnungsindizes und internen Indizes überprüft werden, um sicherzustellen, dass Daten korrekt katalogisiert werden.

Berechtigung

- Die Berechtigungs-ID muss mindestens über Leseberechtigung für das Textindexverzeichnis verfügen.
- Ein Benutzer, der Operationen auf Datenbankinstanzebene durchführen kann, verfügt über die erforderlichen Berechtigungen zur Ausführung dieses Dienstprogramms.

Syntax



Index-Pfad-Klausel:

`--p` *absoluter Indexpfad* `--i` *Index-ID*

Index-Datenbank-Klausel:

`--indschema` *Indexschemaname* `--indname` *Indexname* `--dbname` *Datenbankname*

Tabellen-Spalten-Datenbank-Klausel:

`--tabschema` *Tabellenschemaname* `--tablename` *Tabellenname*

`--colname` *Spaltenname* `--dbname` *Datenbankname*

Parameter

`-p` *absoluter Indexpfad*

Gibt den absoluten Pfad eines Indexverzeichnisses an.

`-i` *Index-ID*

Gibt die ID des zu prüfenden Index an.

`-indschema` *Indexschemaname*

Gibt das Indexschema an, für das der NSE-Index überprüft wird.

`-indname` *Indexname*

Gibt den Indexnamen an, für den der NSE-Index überprüft wird.

`-dbname` *Datenbankname*

Gibt den Namen der Datenbank an, in der der NSE-Index erstellt wird.

`-tabschema` *Tabellenschemaname*

Gibt das Tabellenschema an, dem der zu überprüfende NSE-Index zugeordnet ist.

`-tablename` *Tabellenname*

Gibt den Tabellennamen an, dem der zu überprüfende NSE-Index zugeordnet ist.

-colname *Spaltenname*

Gibt den Namen der Tabellenspalte an, für die der zu überprüfende NSE-Index erstellt wird.

-v Zeigt die ausführlich Ausgabe des Tools an.

-deepCheck

Prüft sowohl Net Search Extender-Zuordnungsindizes als auch interne Indizes, um sicherzustellen, dass sie synchron sind. Wenn Sie das Dienstprogramm ohne den Parameter **-deepCheck** ausführen und das Dienstprogramm meldet, dass die Indizes gültig sind, dass jedoch eine der nachfolgend aufgeführten Bedingungen zutrifft, führen Sie das Dienstprogramm mit dem Parameter **-deepCheck** erneut aus:

- Suchabfragen geben Fehler zurück, jedoch ohne bestimmte Fehlercodes oder Erläuterungen.
- In den Suchergebnissen sind keine neu hinzugefügten oder aktualisierten Textdokumente berücksichtigt.
- Die Indexaktualisierung schlägt fehl.
- Das Ereignisprotokoll enthält Kernelfehlernachrichten.

Wenn Sie das Dienstprogramm ohne den Parameter **-deepCheck** ausführen und das Dienstprogramm meldet, dass die Indizes ungültig sind, ist es nicht erforderlich, das Dienstprogramm mit dem Parameter **-deepCheck** erneut auszuführen.

-h Zeigt Hilfe für das Dienstprogramm an.

Dieses Dienstprogramm benötigt eine erhebliche Menge an Systemressourcen; abhängig von der Größe des Textindex kann die Ausführung des Dienstprogramms viel Zeit in Anspruch nehmen.

Beispiel

Beispiel 1: Geben Sie den Befehl **checknseindex** ein, um den Status eines gültigen Textindex zu prüfen:

```
C:\SQLLIB\bin\checknseindex.exe -i IX335811 -p "C:\myTextIndexes\NODE0000"  
CTE5265I Das NSE-Indexprüfungsprogramm hat festgestellt, dass der angegebene Index,  
"IX335811", gültig ist.
```

Hinweise zur Verwendung

- Das Net Search Extender-Dienstprogramm zur Indexprüfung befindet sich im Verzeichnis SQLLIB/bin. Verwenden Sie **checknseindex.exe** für die Ausführung des Dienstprogramms unter Windows-Betriebssystemen. Verwenden Sie **checknseindex** für die Ausführung unter AIX-Betriebssystemen.
- Modifizieren Sie die folgende SQL-Beispielanweisung, um die ID eines Index auf Ihrem System abzufragen:

```
select a.indexidentifier, indexdirectory  
from db2ext.ttextcolumns a, db2ext.ttextindexes b  
where a.indexidentifier = b.indexidentifier  
and a.tablename = 'MYTAB'  
and a.schemaname = 'MYSCHEMA' and a.columnname = 'MYDATA'"
```

- In einer partitionierten Umgebung müssen Sie das Dienstprogramm in allen betreffenden Knoten ausführen.

Kapitel 37. Löschen eines Textindex

Wenn Sie beabsichtigen, keine Textsuchvorgänge mehr in einer Textspalte durchzuführen, setzen Sie den Befehl **DROP INDEX FOR TEXT** ab.

Vorbereitende Schritte

Die Berechtigungs-ID der Anweisung muss mindestens eine der folgenden Berechtigungen umfassen:

- Das Zugriffsrecht CONTROL für die Tabelle, für die der Index definiert ist.
- Die Berechtigung DBADM.

Beispiel

```
db2text DROP INDEX comment FOR TEXT
```

Wenn Sie einen Textindex löschen, werden dabei auch die folgenden Tabellen und Sichten gelöscht:

- Die Protokolltabelle und -sicht des Index
- Die Ereignistabelle und -sicht des Index
- Die Trigger für die Protokolltabelle (falls vorhanden)
- Die textverwaltete Zwischenspeichertabelle und Sicht (falls vorhanden)

Anmerkung: Löschen Sie immer die Indizes für die Tabelle, bevor Sie die Tabelle löschen. Wenn Sie erst die Tabelle löschen, werden alle textverwalteten Zwischenspeichertabellen (falls vorhanden) ebenfalls gelöscht, aber die Indizes mit ihren Verwaltungstabellen und Sichten sind weiterhin vorhanden.

Kapitel 38. Anzeigen eines Textindexstatus

Verwenden Sie die Net Search Extender-Katalogsichten zum Abrufen von Informationen zu den aktuellen Textindizes in der Datenbank.

Beispiel

Um zum Beispiel aktuelle Datenbankstandardwerte abzurufen, geben Sie folgenden Befehl ein:

```
db2 "select * from db2ext.dbdefaults"
```

Zum Abrufen von Informationen über die aktuell verfügbaren Indizes, die entsprechenden Tabellen und die Anzahl der indexierten Dokumente verwenden Sie folgenden Befehl:

```
db2 "select indschema, indname, tabschema, tabname, number_docs
      from db2ext.textindexes"
```

Verwenden Sie diesen Befehl zum Abrufen von Informationen über die Formate eines bestimmten Index:

```
db2 "select format, modelname from db2ext.textindexformats where
      indschema = 'DB2EXT' and indname = 'TITLE'"
```

Wenn COMMITCOUNT nicht definiert wurde, wird der Parameter NUMBER_DOCS aus der Sicht 'db2ext.textindexes' während eines aktiven Aktualisierungsprozesses nicht aktualisiert. Geben Sie den folgenden Befehl ein, um die aktuelle Anzahl der Dokumente anzuzeigen, die während des Aktualisierungsprozesses aktualisiert wurden:

```
db2text CONTROL LIST ALL LOCKS FOR DATABASE sample INDEX db2ext.title
```

Kapitel 39. Backup und Restore von Textindizes

Sie müssen die Net Search Extender-Services stoppen, bevor Sie für aktivierte Datenbanken und Textindizes ein Backup oder einen Restore durchführen.

Vorgehensweise

- Führen Sie die folgenden Schritte aus, um ein Backup der aktivierten Datenbanken und der von Net Search Extender erstellten Textindizes durchzuführen:
 1. Ermitteln Sie, welche Indizes von Net Search Extender erstellt wurden und wo diese gespeichert sind. Rufen Sie eine Anweisung SELECT für die Sicht 'db2ext.textindexes' auf:

```
db2 "select indschema, indname, indexdirectory from db2ext.textindexes"
```
 2. Stellen Sie sicher, dass zurzeit keine Indexaktualisierung ausgeführt wird, und stoppen Sie die Net Search Extender-Services. Setzen Sie den folgenden Befehl ab:

```
db2text stop
```
 3. Führen Sie ein Backup der Indexverzeichnisse und ihrer Unterverzeichnisse durch, nachdem Sie ein Backup für die Datenbank durchgeführt haben.
 4. Starten Sie die Net Search Extender-Services erneut. Setzen Sie den folgenden Befehl ab:

```
db2text start
```
- Führen Sie die folgenden Schritte aus, um einen Restore der aktivierten Datenbanken und der von Net Search Extender erstellten Textindizes durchzuführen:
 1. Stoppen Sie Net Search Extender. Setzen Sie den folgenden Befehl ab:

```
db2text stop
```
 2. Stellen Sie die Backupkopien der Indexverzeichnisse in ihrem ursprünglichen Pfad wieder her.
 3. Starten Sie Net Search Extender erneut. Setzen Sie den folgenden Befehl ab:

```
db2text start
```

Kapitel 40. Entfernen von Dateien aus dem Verzeichnis /tmp

Bestimmte Dateien dürfen nicht aus dem Verzeichnis /tmp gelöscht werden, während die Net Search Extender-Services ausgeführt werden.

Während der Ausführung der Net Search Extender-Services müssen Sie sicherstellen, dass geplante Jobs, die den Inhalt von /tmp löschen, diese Dateien nicht unbeabsichtigt entfernen. Die folgenden Dateien müssen im Verzeichnis /tmp vorhanden sein und dürfen während der Ausführung der Net Search Extender-Services nicht gelöscht werden:

- Semaphore und gemeinsam benutzte Speicherdateien:

```
instanzeigner.TEXT.0000.LATCH  
instanzeigner.TEXT.0000  
instanzeigner.CACHE.0000  
instanzeigner.SCHEDULER.LATCH  
instanzeigner.DEMON.SEM  
instanzeigner.DEMON.MEM
```

Anmerkung: In einer Umgebung mit partitionierten Datenbanken würden entsprechend den jeweiligen Knoten zusätzliche Dateien mit ähnlichen Namen vorhanden sein: *instanzeigner.TEXT.0001.LATCH*, *instanzeigner.TEXT.0001*, *instanzeigner.CACHE.0001* usw.

- Bei der Erstellung eines Index können sich auch Dateien wie die folgenden im Verzeichnis /tmp befinden, wenn der Cache temporär ist:

```
datenbankname.IX123456  
datenbankname.IX123456.data0
```

Teil 11. Methoden zum Durchsuchen von Text

Net Search Extender ermöglicht die Verwendung von skalaren SQL-Funktionen, einer Suchfunktion mit gespeicherten Prozeduren und einer SQL-Tabellenwertfunktion für das Durchsuchen von Texten.

Net Search Extender bietet folgende Methoden für die Textsuche:

Skalare SQL-Suchfunktionen

Unterabfragen für die Textsuche können in SQL-Abfragen integriert werden. Net Search Extender stellt die skalaren SQL-Suchfunktionen als Erweiterung für die verfügbaren SQL-Funktionen bereit. Wenn Sie in SQL-Abfragen Unterabfragen für die Textsuche integrieren, können Net Search Extender-Suchfunktionen mit der DB2-XQuery-Verarbeitung kombiniert werden. Textsuchabfragen für XML-Dokumente können in der XQuery-Eingabefunktion `db2-fn:sqlquery()` verwendet werden; sie ermöglichen eine direkte Verarbeitung der resultierenden XML-Dokumente mit XQuery.

Suche mit gespeicherten Prozeduren

Diese Suchoption gibt Ihnen die Möglichkeit, vordefinierte und im Cache gespeicherte Ergebnistabellen zurückzugeben.

SQL-Tabellenwertfunktion

Diese Suchfunktion kann in ähnlicher Weise wie die Suche mit gespeicherten Prozeduren verwendet werden.

Für skalare SQL-Suchfunktionen beschreibt dieser Abschnitt folgende Bereiche:

- Suchen nach Text mithilfe der Funktionen CONTAINS, NUMBEROFMATCHES und SCORE.
Eine ausführliche Beschreibung der Syntax finden Sie in Kapitel 60, „Skalare SQL-Suchfunktion und SQL-Tabellenwertfunktion“, auf Seite 267.
- Angeben von Suchargumenten anhand von Beispielen mit der Funktion CONTAINS.
Eine vollständige Beschreibung der Syntax finden Sie in Kapitel 59, „Syntax für Suchargumente“, auf Seite 257.

Für Suchoperationen mit einer gespeicherten Prozedur beschreibt dieser Abschnitt folgende Bereiche:

- Suchen nach Text mit einer gespeicherten Prozedur.
- Eine Beschreibung der Parameter bei der Angabe von Suchargumenten finden Sie in Kapitel 59, „Syntax für Suchargumente“, auf Seite 257.

Für die SQL-Tabellenwertfunktion beschreibt dieser Abschnitt die folgenden Bereiche:

- Suchen nach Text mit der SQL-Tabellenwertfunktion und der Hervorhebungsfunktion (HIGHLIGHT).
Eine Beschreibung der Syntax finden Sie in Kapitel 60, „Skalare SQL-Suchfunktion und SQL-Tabellenwertfunktion“, auf Seite 267.
- Eine Beschreibung der Parameter bei der Angabe von Suchargumenten finden Sie in Kapitel 59, „Syntax für Suchargumente“, auf Seite 257.

Darüber hinaus finden Sie Informationen zu Aspekten der Suchleistung, die unter Umständen zu berücksichtigen sind.

Stellen Sie vor dem Suchen sicher, dass die entsprechenden Indexierungsschritte, die in Teil 9, „Entwicklung: Erstellen und Pflegen eines Textindex“, auf Seite 91 beschrieben werden, für die verschiedenen Datentypen durchgeführt wurden.

Kapitel 41. Suchen nach Text mit skalaren SQL-Suchfunktionen

Die skalaren SQL-Suchfunktionen können auf verschiedene Weise durch die Verwendung der Funktionen CONTAINS, NUMBEROFMATCHES und SCORE genutzt werden.

Mithilfe von Beispielen wird in diesem Abschnitt die Verwendung von skalaren SQL-Suchfunktionen beschrieben:

- Verwenden der Funktion CONTAINS zum Absetzen einer Abfrage.
- Verwenden der Funktion NUMBEROFMATCHES zum Ermitteln der Anzahl an Übereinstimmungen des Suchbegriffs in einem Textdokument.
- Verwenden der Funktion SCORE zum Abrufen der Relevanz eines gefundenen Textdokuments.

Eine Beschreibung der Syntax finden Sie in Kapitel 60, „Skalare SQL-Suchfunktion und SQL-Tabellenwertfunktion“, auf Seite 267.

Absetzen einer Abfrage

Durch das Absetzen einer Abfrage der Funktion CONTAINS wird nach Text in Spalten gesucht, die in Tabellen gefunden werden.

Beispiel

In diesem Beispiel wird veranschaulicht, wie die Funktion CONTAINS nach Text in der Spalte comment der Tabelle texttab sucht. Die Funktion gibt den Wert 1 zurück, wenn der Text das Suchargument erfüllt; ansonsten wird der Wert 0 zurückgegeben.

```
SELECT AUTHOR,TITLE
       FROM DB2EXT.TEXTTAB
WHERE CONTAINS(COMMENT, '"book"') = 1
```

In diesem Beispiel wird nach dem Begriff book in der Spalte COMMENT gesucht.

Die Suche nach "" wird nicht unterstützt. Die Verwendung von zwei aufeinander folgenden Anführungszeichen in einem Suchbegriff führt zu einer Syntaxfehlermeldung. Ebenso wird bei der Verwendung eines Zeilenvorschubzeichens im Suchbegriff ein Abfragesyntaxfehler ausgegeben.

Anmerkung:

Wenn Sie wissen, dass durch die Textsuche alleine eine sehr große Ergebnismenge zurückgegeben wird, sollten einschränkende Suchbedingungen hinzugefügt werden. Beispiel:

```
SELECT AUTHOR,TITLE
       FROM db2ext.texttab
WHERE CONTAINS(COMMENT, '"book"') = 1 AND PRICE < 20
```

Suchen und Ermitteln der Anzahl gefundener Übereinstimmungen

Mithilfe der Funktion NUMBEROFMATCHES können Sie eine Suche durchführen und die Anzahl der gefundenen Übereinstimmungen ermitteln.

Informationen zu diesem Vorgang

Mithilfe der Funktion NUMBEROFMATCHES können Sie ermitteln, wie oft der Suchbegriff in jedem Textdokument gefunden wurde.

```
SELECT AUTHOR, TITLE, NUMBEROFMATCHES(COMMENT, 'book')
      FROM DB2EXT.TEXTTAB WHERE
      NUMBEROFMATCHES(COMMENT, 'book') > 0
```

NUMBEROFMATCHES gibt für jede Zeile einen ganzzahligen Wert zurück.

Suchen und Ermitteln der Quote eines gefundenen Textdokuments

Durch SCORE wird eine positive Zahl zurückgegeben, die angibt, wie gut das Dokument dem Suchbegriff im Verhältnis zu den anderen gefundenen Dokumenten im gleichen Index entspricht. Der Wert wird auf der Basis der Anzahl an Übereinstimmungen, die im Dokument gefunden werden, im Verhältnis zur Größe des Dokuments berechnet.

Beispiel

Im folgenden Beispiel kann die Quote eines gefundenen Dokuments unter Verwendung der Funktion SCORE ermittelt werden:

```
WITH TEMPTABLE(docid,score)
      AS (SELECT docid,
                SCORE(COMMENT, 'book')
          FROM DB2EXT.TEXTTAB)
SELECT *
      FROM TEMPTABLE
     WHERE score > 0
     ORDER BY score ASC
```

Die Funktion SCORE gibt einen DOUBLE-Wert zwischen 0 und 1 zurück.

Die durch SCORE zurückgegebenen Werte sind nur dann aussagefähig, wenn sie mit anderen SCORE-Werten verglichen werden, die für den gleichen Index zurückgegeben werden. Die Werte können nicht mit Quoten verglichen werden, die für andere Indizes zurückgegeben werden.

Anmerkung: Die Suchfunktionen CONTAINS, SCORE und NUMBEROFMATCHES können nicht für Indizes verwendet werden, die für Sichten erstellt wurden.

Die SCORE-Werte variieren je nach der DB2-Datenbankumgebung:

- In einer Umgebung ohne partitionierte Datenbanken befinden sich alle Dokumente in einer einzigen Tabelle. Der Wert SCORE basiert auf einer einzigen Tabelle sowie der Beziehung eines Dokuments zu allen anderen Dokumenten innerhalb der Tabelle.
- In einer Umgebung mit partitionierten Datenbanken befinden sich die Dokumente in unterschiedlichen Partitionen. Bei der Indexierung werden nur die Dokumente zum Erstellen des Textindex verwendet, die sich in jeweils einer Partition lokal befinden. In diesem Fall basiert der Wert SCORE auf der Beziehung der Dokumente zu allen anderen Dokumenten in nur einer der vielen Partitionen.

Kapitel 42. Angeben von SQL-Suchargumenten

Die Funktionen CONTAINS, NUMBEROFMATCHES und SCORE verwenden Suchargumente. Dieser Abschnitt der Dokumentation zeigt anhand der Funktion CONTAINS verschiedene Beispiele für Suchargumente in Net Search Extender-Funktionen.

Eine vollständige Beschreibung der Syntax finden Sie in „Syntax für Suchargumente“ auf Seite 257.

Suchen nach Begriffen in beliebiger Reihenfolge

Sie können mehr als einen Begriff in ein Suchargument einschließen. Eine Möglichkeit, mehrere Suchbegriffe zu kombinieren, besteht darin, sie mit Kommas zu verbinden.

Beispiel

Beispiel:

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
      WHERE CONTAINS(COMMENT,
                    '("kid", "dinosaur")') = 1
```

Diese Form des Sucharguments sucht nach Textelementen, die einen beliebigen Suchbegriff enthalten; die Reihenfolge ist dabei beliebig. In logischen Begriffen ausgedrückt bedeutet dies, dass die Suchbegriffe durch einen impliziten Operator OR verbunden werden.

Suchen mit den Booleschen Operatoren AND und OR

Suchbegriffe können mithilfe der Booleschen Operatoren "&" (AND) und "|" (OR) mit anderen Suchbegriffen kombiniert werden.

Informationen zu diesem Vorgang

Suchbegriffe können mithilfe der Booleschen Operatoren "&" (AND) und "|" (OR) mit anderen Suchbegriffen kombiniert werden:

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
      WHERE CONTAINS(COMMENT,
                    '"author" | "pulitzer"') = 1
```

Sie können auch mehrere Begriffe mit Booleschen Operatoren kombinieren:

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
      WHERE CONTAINS(COMMENT,
                    '"author" | "pulitzer" & "book"') = 1
```

Wenn Sie mehrere Boolesche Operatoren verwenden, werden diese von links nach rechts ausgewertet. Der logische Operator AND (&) bindet jedoch wie bei der regulären Booleschen Logik stärker als der logische Operator OR (|). Diese Bewertung wird im folgenden Beispiel, das keine runden Klammern enthält, gezeigt:

```
"book" & "pulitzer" | "year" & "author"
```

Net Search Extender bewertet die Booleschen Operatoren folgendermaßen:

```
("book" & "pulitzer") | ("year" & "author")
```

Wenn Sie eine andere Bewertungsreihenfolge der Booleschen Operatoren erzwingen möchten, müssen runde Klammern eingefügt werden:

```
"book" & ("pulitzer" | "year") & "author"
```

Sie können Boolesche Operatoren auch mit Suchbegriffen kombinieren, die mit einem Komma als Trennzeichen verkettet werden:

```
("author", "pulitzer") & "book"
```

In diesem Fall wird das Komma als Boolescher Operator OR interpretiert:

```
("author" | "pulitzer") & "book"
```

Suchen mit dem Booleschen Operator NOT

Sie können den Booleschen Operator NOT verwenden, um bestimmte Textdokumente von der Suche auszuschließen.

Beispiel

Beispiel:

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
      WHERE CONTAINS(COMMENT,
                    '("author", "pulitzer") & NOT "book"') = 1
```

In diesem Beispiel werden alle Textdokumente, die den Begriff "book" enthalten, von den Suchergebnissen für die Begriffe "author" oder "pulitzer" ausgeschlossen.

Suchen nach groben Übereinstimmungen

Eine *Suche nach groben Übereinstimmungen* zielt auf Wörter ab, die ähnlich wie der Suchbegriff geschrieben sind.

Beispiel

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
      WHERE CONTAINS(COMMENT,
                    'fuzzy form of 80 "pullitzer"') =1
```

Diese Suchoperation könnte ein Vorkommen des fehlerhaft geschriebenen Wortes Pulitzer finden.

Der Übereinstimmungsgrad, in diesem Beispiel "80", gibt den erforderlichen Wert für die Genauigkeit an. Verwenden Sie die Suche nach grober Übereinstimmung, wenn anzunehmen ist, dass das Dokument Rechtschreibfehler enthält. Dies ist häufig der Fall, wenn das Dokument mithilfe einer Einheit zur optischen Zeichenerkennung oder zur Spracheingabe erstellt wird. Es werden Werte zwischen 1 und 100 verwendet, um den Grad der groben Übereinstimmung anzugeben; dabei steht 100 für eine exakte Übereinstimmung, und jeder Wert zwischen 80 und 1 bedeutet eine zunehmend grobere Übereinstimmung.

Anmerkung: Wenn die Suche nach grober Übereinstimmung nicht das gewünschte Maß an Genauigkeit erbringt, sollten Sie nach einzelnen Teilen des Suchbegriffs suchen, indem Sie Platzhalterzeichen verwenden.

Suchen nach Teilen eines Begriffs (Platzhalterzeichen)

Platzhalterzeichen stellen eine Methode dar, um eine Suche flexibler zu gestalten. Sie erhöhen die Anzahl von Textdokumenten, die bei einer Suche gefunden werden. Net Search Extender arbeitet mit zwei Platzhalterzeichen: Prozentzeichen (%) und Unterstreichungszeichen (_).

Informationen zu diesem Vorgang

Net Search Extender verwendet diese Platzhalterzeichen auf die gleiche Weise wie das DB2-Vergleichselement LIKE.

- % steht für eine beliebige Anzahl von Platzhalterzeichen.

Das folgende Beispiel zeigt das Zeichen % in der Verwendung als Platzhalterzeichen innerhalb eines Suchbegriffs:

```
SELECT AUTHOR, TITLE
       FROM DB2EXT.TEXTTAB
       WHERE CONTAINS(COMMENT, 'th%er') = 1
```

Mit diesem Suchbegriff wird nach Textdokumenten gesucht, die das Wort "thriller", "throttle" und "thread-splitter" enthalten.

- _ steht für ein Zeichen in einem Suchbegriff.

Mit diesem Suchbegriff wird nach Textdokumenten gesucht, die das Wort "thriller" enthalten.

```
SELECT AUTHOR, TITLE
       FROM DB2EXT.TEXTTAB
       WHERE CONTAINS(COMMENT, 'th_i1ler') = 1
```

Sie können in einer Wortfolge mehr als ein Platzhalterzeichen verwenden (mehrere Wörter in einer Wortfolge können ein Platzhalterzeichen enthalten); allerdings kann es sich bei den aus einer Platzhalterzeichenerweiterung resultierenden Begriffen nur um einzelne Begriffe und nicht um Begriffe aus mehreren Wörtern handeln. Beispiel: Der Platzhalterausdruck "th%er" stimmt nicht mit der Wortfolge "the caller" überein.

Sie sollten Platzhalterzeichen sparsam verwenden, da sie Ihre Ergebnisliste erheblich vergrößern und dadurch die Leistung mindern und zudem unerwünschte Suchergebnisse liefern können.

Beachten Sie, dass Sie eine unscharfe Suche oder eine Thesaurussuche nicht mit einer Suche mit Platzhalterzeichen kombinieren können. Stellen Sie darüber hinaus sicher, dass die Platzhalterzeichen innerhalb des impliziten oder expliziten Suchparameters EXPANSION LIMIT *anzahl* erweiterbar sind. Einzelheiten zum Suchparameter EXPANSION LIMIT *anzahl* finden Sie in „Suchparameter“ auf Seite 260.

Suchen nach Begriffen, die ein Platzhalterzeichen enthalten

Wenn Sie nach einem Begriff suchen möchten, der das Zeichen '%' oder das Zeichen '_' enthält, müssen Sie dem Zeichen ein so genanntes *Escapezeichen* voranstellen. Sie müssen das Escapezeichen in der Abfrage mit dem Schlüsselwort ESCAPE angeben.

Beispiel

Im folgenden Beispiel ist das Escapezeichen ein "!":

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
     WHERE CONTAINS(COMMENT,
                    "'100!%' ESCAPE '!') = 1
```

Suchen nach Begriffen in einer festen Reihenfolge

Sie können nach Begriffen in einer festen Reihenfolge suchen, wenn Sie nach "primary key" suchen.

Beispiel

Wenn Sie nach dem Begriff "primary key" suchen, finden Sie die beiden Begriffe nur, wenn Sie direkt benachbart sind und in der nachfolgend gezeigten Reihenfolge vorkommen:

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
     WHERE CONTAINS(COMMENT, '"primary key"') = 1
```

Suchen nach Begriffen im gleichen Satz oder Absatz

Net Search Extender hat nur eine begrenzte Fähigkeit, im selben Satz oder Absatz nach Begriffen zu suchen.

Beispiel

Das folgende Beispiel zeigt ein Suchargument, das nach Textdokumenten sucht, in denen der Suchbegriff 'web' im gleichen Satz vorkommt wie der Begriff 'disk':

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
     WHERE CONTAINS(COMMENT,
                    '"web" IN SAME SENTENCE AS "disk"') = 1
```

Sie können auch nach mehreren zusammen auftretenden Wörtern suchen. Im nächsten Beispiel wird nach zwei Wortfolgen gesucht, die in demselben Abschnitt auftreten:

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
     WHERE CONTAINS(COMMENT,
                    '"linguistic analysis processing" IN SAME PARAGRAPH AS
                    "search algorithms"') = 1
```

Suchen nach Begriffen in Abschnitten strukturierter Dokumente

Das im vorliegenden Abschnitt beschriebene Beispiel zeigt ein Suchargument, mit dem Textdokumente gesucht werden, in denen der Suchbegriff "IBM" im Unterabschnitt "H2" von strukturierten Dokumenten vorkommt.

Beispiel

Das folgende Beispiel zeigt ein Suchargument, mit dem Textdokumente gesucht werden, in denen der Suchbegriff "IBM" im Unterabschnitt "H2" von strukturierten Dokumenten vorkommt.

```
SELECT CATEGORY, DATE
      FROM DB2EXT.HTMLTAB
     WHERE CONTAINS(HTMLFILE,
                    'SECTIONS ("H2") "IBM"') = 1
```

Beachten Sie, dass für Abschnittsnamen die Groß-/Kleinschreibung zu beachten ist. Stellen Sie sicher, dass der Abschnittsname in der Modelldatei und in der Abfrage identisch sind.

Thesaurussuche

Die Thesaurussuche ist eine leistungsstarke Funktion zur Suchbegriffserweiterung in Net Search Extender. Die zusätzlichen Begriffe, nach denen gesucht wird, werden einem Thesaurus entnommen, den Sie selbst erstellen, sodass Sie die direkte Kontrolle über diese Begriffe haben.

Informationen zu diesem Vorgang

Beispiel: Bei einer Thesaurussuche nach dem Begriff 'database' könnten Sie gleichzeitig Begriffe wie 'repository' und 'DB2' finden, wenn Sie angeben, dass diese Begriffe zusammengehörig sind.

Verwenden Sie diese Art der Suche für bestimmte Interessengebiete, in denen Sie häufig Suchen durchführen, da sie erheblich effektivere Suchergebnisse liefert.

Beispiel

Die folgenden Beispiele veranschaulichen die Syntax zur Verwendung der Thesauruserweiterung.

In diesem Beispiel wird der Begriff "product" erweitert, indem alle zugehörigen Begriffe dieses Begriffs, die im Thesaurus 'nseamplethes' zu finden sind, hinzugefügt werden.

```
SELECT CATEGORY, DATE
      FROM DB2EXT.HTMLTAB
      WHERE CONTAINS(HTMLFILE,
                    'THESAURUS "nseamplethes"
                    EXPAND RELATED
                    TERM OF "product"') = 1
```

Im nächsten Beispiel handelt es sich um den Suchbegriff 'product'. Die Suche wird dann durch alle *Synonyme* des Suchbegriffs erweitert.

```
SELECT CATEGORY, DATE
      FROM DB2EXT.HTMLTAB
      WHERE CONTAINS(HTMLFILE,
                    'THESAURUS "nseamplethes"
                    EXPAND SYNONYM
                    TERM OF "product"') = 1
```

Suchen nach numerischen Attributen

Sie können numerische Attribute, die in einem Textindex gespeichert werden, durchsuchen.

Informationen zu diesem Vorgang

Numerische Attribute, die in einem Textindex gespeichert werden, können mithilfe der folgenden Syntax durchsucht werden:

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
      WHERE CONTAINS(COMMENT,
                    'ATTRIBUTE "PRICE" between 9 and 20') = 1
```

Freitextsuche

Eine *Freitextsuche* ist eine Suche, bei der der Suchbegriff als Text mit freiem Format angegeben wird. Ein Ausdruck oder ein Satz beschreibt in natürlicher Sprache das zu suchende Thema.

Informationen zu diesem Vorgang

In einer Freitextsuche ist die Wortreihenfolge irrelevant. Allerdings muss bei der Freitextabfrage mindestens ein Abfragebegriff in den zu durchsuchenden Dokumenten auftreten.

Beachten Sie, dass Platzhalterzeichen für einzelne Zeichen oder Wörter für Suchzeichenfolgen in einem Freitextsuchargument nicht unterstützt werden.

Beispiel

Beispiel:

```
SELECT AUTHOR, TITLE, SCORE(COMMENT,  
  'IS ABOUT EN_US "something related to dinosaur"')  
  FROM DB2EXT.TEXTTAB  
  WHERE CONTAINS(COMMENT,  
  'IS ABOUT EN_US "something related to dinosaur"') = 1
```

Kapitel 43. Zusätzliche Suchsyntaxbeispiele

Führen Sie das Script **search** aus, das sich im Verzeichnis `sqllib/samples/extenders/db2ext/` befindet, wenn Sie sich mit zusätzlichen Beispielen zur Suchsyntax vertraut machen möchten.

Informationen zu diesem Vorgang

Dieses enthält Beispiele für Net Search Extender-Suchfunktionen, die für die Beispieltabelle ausgeführt werden.

Geben Sie den Befehl wie folgt ein:

```
db2 -tvf search
```

Es ist auch ein Beispiel vorhanden, in dem Sie sehen können, wie XML-Daten abgefragt werden können. Wenn Sie eine Verbindung zur Datenbank hergestellt haben, können Sie für die Daten mithilfe des Befehls **db2 -tvf xmlsearch** Suchoperationen durchführen.

Wenn die Tabelle und die Indizes nicht erstellt wurden, führen Sie eine der folgenden Operationen aus:

- Unter UNIX-Betriebssystemen: **nsesample** im Verzeichnis `ausgangsverzeichnis_des_instanzeigners/sqllib/samples/extenders/db2ext`.
- Unter Windows-Betriebssystemen: `nsesample (.bat)` im Verzeichnis `sqllib\samples\extenders\db2ext`.
- Rufen Sie bei XML-Suchen `xmlsample (.bat)datenbank` auf, um die Datenbank zu füllen, und erstellen und aktualisieren Sie die Indizes.

Kapitel 44. Suchen nach Text mithilfe einer Suche mit gespeicherten Prozeduren

Verwenden Sie die Schnittstelle für die Suche mit gespeicherten Prozeduren, wenn Sie nur eine Untergruppe mit Rangordnung der Ergebnisse der Textsuche und eine hohe Abfrageleistung benötigen.

Verwenden Sie die gespeicherte Prozedur nicht, wenn alle Suchergebnisse benötigt werden oder wenn eine große Anzahl an Dokumenten indexiert werden muss. Der Hauptgrund hierfür ist, dass Teile der Benutzertabelle in den Speicher kopiert werden und somit eine Menge Realspeicher verfügbar sein muss.

Sie können die gespeicherte Prozedur verwenden, um zunächst die Ergebnisse 0 bis 20, dann 21 bis 40 usw. in einer ähnlichen Weise wie bei der Cursornavigation abzurufen. Die Kombination der Cursorfähigkeit mit der Verwendung eines Caches (der bei der Indexierung errechnet wird) ermöglicht eine extrem hohe Suchgeschwindigkeit, besonders deshalb, weil kein Join für die Benutzertabelle erforderlich ist.

Wenn Sie die gespeicherte Prozedur verwenden möchten, berücksichtigen Sie dabei die folgenden Faktoren:

- Die Optionen für den Suchergebniscache wurden im Befehl **CREATE INDEX** angegeben.
- Gegenwärtige und zukünftige Anforderungen an den gemeinsamen Speicher, die möglicherweise Teilaktualisierungen einschließen, wurden vollständig berücksichtigt.
- Der Cache des Index wurde mithilfe des Befehls **db2text activate** aktiviert.
- Damit Sie in einer Umgebung mit partitionierten Datenbanken die Suche mit gespeicherten Prozeduren verwenden können, muss die Tabelle einen Tabellenbereich in einer einzigen Partition verwenden und die Prozedur sollte für dieselbe Partition aufgerufen werden. Andernfalls wird die Suche nicht zugelassen und es werden Fehler zurückgegeben.

Dies ist ein Beispiel für eine Suche mit gespeicherten Prozeduren:

```
db2 "call db2ext.textSearch('\book\'', 'DB2EXT', 'COMMENT', 0, 2, 1, 1, '?', '?")"
```

Der erste Parameter ist der Suchbegriff. Die Syntax für den Suchbegriff ist mit der Syntax in den SQL-Skalarfunktionen identisch. Bei den nächsten Parametern handelt es sich um das Indexschema und den Indexnamen. Wenn Sie den Namen nicht in Anführungsstriche gesetzt haben, wird er in Großbuchstaben umgesetzt. Die folgenden beiden numerischen Argumente geben den Ausgangspunkt des Ergebnis-sektors und die Anzahl an Ergebnissen im Sektor an. Die beiden nächsten ganzzahligen Werte geben an, ob Informationen zur Quote und zur Trefferanzahl angefordert werden. Die letzten beiden Werte sind die Funktionsrückgabewerte.

Anmerkung: Wenn Sie größere Ergebnismengen abfragen, benötigen Sie einen Benutzertabellenbereich. Wenn keiner verfügbar ist, erstellen Sie einen Tabellenbereich. Im folgenden Beispiel wird ein Tabellenbereich für eine UNIX-Plattform erstellt:

```
db2 "create user temporary tablespace tempts managed by system
      using ('/work/tempts.ts')"
```

Kapitel 45. Suchen nach Text mit einer SQL-Tabellenwertfunktion

Verwenden Sie die SQL-Tabellenwertfunktion, wenn Sie nicht alle Suchergebnisse benötigen und Sie nicht über ausreichend Speicher für die Verwendung eines Cachindex verfügen, wie er in der Suche mit gespeicherten Prozeduren verwendet wird.

Es sind zwei SQL-Tabellenwertfunktionen verfügbar, die beide `db2ext.textsearch` heißen. Hierbei verfügt eine der beiden über zusätzliche Parameter für die Verwendung mit der Funktion `db2ext.highlight`.

Die SQL-Tabellenwertfunktion stellt Ihnen die gleiche Cursorschnittstelle zum Zugreifen nur auf Ausschnitte des Ergebnisses zur Verfügung wie die gespeicherte Prozedur. Sie müssen jedoch die Ergebnisse noch mit der Benutzertabelle verknüpfen. Dies wird im folgenden Beispiel demonstriert:

```
db2 "select docid , author, score from TABLE(db2ext.textsearch('\\"book\\" ',
      'DB2EXT','COMMENT',3,2,cast(NULL as integer))) as t, db2ext.texttab u
      where u.docid = t.primkey"
```

Die folgenden Werte könnten Sie aus der SQL-Tabellenwertfunktion zurückgeben:

```
--> primaryKey <typ eines einzelnen primärschlüssels>
der Primärschlüssel
```

```
--> score          DOUBLE
der Quotenwert des gefundenen Dokuments
```

```
--> NbResults      INTEGER
die Gesamtanzahl gefundener Ergebnisse (gleicher Wert für alle Zeilen)
```

```
--> numberOfMatches INTEGER
die Anzahl der Treffer im Dokument
```

Anmerkung:

- Nur eine einzige Primärschlüsselspalte ist zulässig.
- Damit Sie in einer Umgebung mit partitionierten Datenbanken die Suche mit der SQL-Tabellenwertfunktion ausführen können, muss die Tabelle einen Tabellenbereich in einer einzigen Partition verwenden und die Funktion sollte für dieselbe Partition aufgerufen werden. Andernfalls wird die Suche nicht zugelassen und es werden Fehler zurückgegeben.

Verwenden der Hervorhebungsfunktion

Zur Verwendung der SQL-Tabellenwertfunktion `db2ext.highlight` ist die Funktion `db2ext.textsearch` mit den zusätzlichen Parametern **numberOfHits** und **hitInformation** erforderlich.

Informationen zu diesem Vorgang

Die Hervorhebungsfunktion darf nicht verwendet werden, wenn Sie einen Index mithilfe der Umsetzungsfunktion erstellt haben, deren Implementierung sich ändern kann. Andernfalls sind die zurückgegebenen hervorgehobenen Positionsdaten möglicherweise nicht korrekt, da eine Abweichung zwischen den positionsgebun-

denen Informationen zum Zeitpunkt der Indexierung und zum Suchzeitpunkt besteht.

Beispiel

In diesem Beispiel wird die Funktion `db2ext.highlight` aufgerufen, um das gesamte Dokument anzuzeigen, ohne dabei Fundstellen hervorzuheben, die mithilfe der Funktion `db2ext.textsearch` gefunden wurden.

```
select p.docid,
       db2ext.highlight(p.comment, t.hitinformation, ' WINDOW_NUMBER = 0 ')
       as highlight
from DB2EXT.TEXTTAB p,
     table (db2ext.textsearch("bestseller" | "peacekeeping" | "soldiers"
                             | "attention", 'DB2EXT', 'COMMENT', 0, 20,
                             cast(NULL as INTEGER), 10)) t
where p.docid = t.primkey and p.docid = 2
```

Die Abfrage gibt das folgende Ergebnis zurück:

DOCID HIGHLIGHT

```
2      A New York Times bestseller about peacekeeping soldiers called
      'Keepers' who devise a shocking scheme to get the worlds
      attention after their tour of duty ends.
```

1 record(s) selected.

In diesem Beispiel wird die Funktion `db2ext.highlight` aufgerufen, um das gesamte Dokument anzuzeigen, wobei die Fundstellen hervorgehoben werden, die mithilfe der Funktion `db2ext.textsearch` gefunden wurden.

```
select p.docid,
       db2ext.highlight(p.comment, t.hitinformation, ' WINDOW_NUMBER = 0,
       TAGS = ("<bf>", "</bf>" ) ') as highlight
from DB2EXT.TEXTTAB p,
     table (db2ext.textsearch("bestseller" | "peacekeeping" | "soldiers"
                             | "attention", 'DB2EXT', 'COMMENT', 0, 20,
                             cast(NULL as INTEGER), 10)) t
where p.docid = t.primkey and p.docid = 2
```

Das Suchargument gibt folgendes Ergebnis zurück:

DOCID HIGHLIGHT

```
2      A New York Times <bf>bestseller</bf> about <bf>peacekeeping</bf>
<bf>soldiers</bf> called 'Keepers' who devise a shocking scheme to
get the worlds <bf>attention</bf> after their tour of duty ends.
```

1 record(s) selected.

In diesem Beispiel wird die Funktion `db2ext.highlight` so aufgerufen, dass maximal 10 Ausschnitte (Fenster) des Dokuments angezeigt werden. Dabei ist das Fenster 24 Zeichen groß, was ungefähr 12 Byte an Daten auf jeder Seite des Treffers entspricht. Zusätzlich werden durch die Funktion `db2ext.textsearch` gefundene Treffer hervorgehoben.

```
select p.docid,
       db2ext.highlight(p.comment, t.hitinformation, ' WINDOW_NUMBER = 10,
       WINDOW_SIZE = 24, TAGS = ("<bf>", "</bf>" ) ') as highlight
from DB2EXT.TEXTTAB p,
     table (db2ext.textsearch("bestseller" | "peacekeeping" | "soldiers"
                             | "attention", 'DB2EXT', 'COMMENT', 0, 20,
                             cast(NULL as INTEGER), 10)) t
where p.docid = t.primkey and p.docid = 2
```

Das Suchargument gibt folgendes Ergebnis zurück:

DOCID HIGHLIGHT

```
2   York Times <bf>bestseller</bf> about <bf>peacekeeping</bf> ...
    <bf>peacekeeping</bf> <bf>soldiers</bf> called 'Keepers' ... the
    worlds <bf>attention</bf> after their
```

1 record(s) selected.

Die erste Fundstelle ist **bestseller**. Dieser Treffer bestimmt das erste Fenster. Der zweite Treffer, **peacekeeping**, ist nur 8 Byte vom ersten Treffer entfernt und wird vollständig in das erste Fenster aufgenommen. Der dritte Treffer, **soldiers**, befindet sich außerhalb des ersten Fensters und bestimmt deshalb ein neues Fenster. Da der zweite Treffer, **peacekeeping**, nur 2 Byte von der linken Seite des Treffers **soldiers** entfernt ist, wird er auch in das zweite Fenster aufgenommen und hervorgehoben. Der vierte Treffer, **attention**, befindet sich außerhalb des zweiten Fensters und bestimmt daher ein neues Fenster. Da die Größe dieses Fensters keinen vorherigen oder zusätzlichen Treffer beinhaltet, sind nur Daten aus der Umgebung des Treffers in dem Fenster enthalten.

Da WINDOW_SEPARATOR nicht angegeben wurde, wird außerdem das Standardfenstertrennzeichen ' ... ' verwendet, um die drei Trefferfenster voneinander zu trennen.

Anmerkung: Um eine möglichst hohe Leistung bei der Verwendung der Funktion db2ext.highlight sicherzustellen, sollte der Benutzer die Suchergebnisse in der Tabellenwertfunktion db2ext.textsearch begrenzen.

Kapitel 46. Suchen in mehreren Spalten

Wenn ein Textindex für mehr als eine Spalte erstellt werden muss, besteht die einfachste Möglichkeit zur Ausführung dieses Arbeitsschrittes in der Verwendung der skalaren SQL-Suchfunktion und im Kombinieren der Suchoperationen für diese Spalten.

Beispiel

Dieser Vorgang ist im folgenden Beispiel dargestellt:

```
SELECT AUTHOR,TITLE
      FROM DB2EXT.TEXTTAB
      WHERE CONTAINS(COMMENT,
                    '"book"')=1 and CONTAINS(AUTHOR,'"Mike"')=1
```

Bei einer Tabellenwertfunktion ist dies schwieriger, da aus Gründen der Systemleistung möglicherweise die Union-Verknüpfung der zurückgegebenen Tabellen verwendet werden muss. Eine weitere Möglichkeit besteht bei Tabellenwertfunktionen in der Verwendung einer Sicht und der Kombination Ihrer Tabellenspalten in einer Sichtspalte, um einen einzigen Textindex in dieser Sichtspalte zu erstellen. Auf diese Weise vermeiden Sie zwei separate Textsuchaufrufe.

Die Kombination der Textspalten führt möglicherweise zu einer Verbesserung der Systemleistung. Ob dies zutrifft, hängt jedoch in hohem Maße von den individuellen Suchanforderungen ab.

Kapitel 47. Verwenden der Textsuche in Outer Joins

Sie können die Suchfunktion `CONTAINS()` verwenden, die innerhalb einer Outer Join-Abfrage verwendet wird.

Bei einer Outer Join-Abfrage mit der Suchfunktion `CONTAINS()` schlägt die Abfrage möglicherweise fehl, und es wird der Ursachencode CTE0129 Nullwerte sind für die Übergabe als Parameter nicht zulässig zurückgegeben, es sei denn, das Vergleichselement `CONTAINS()` verweist auf die Spalte einer Tabelle auf der Tupelseite des Outer Joins.

Beispiel: T1 ist die Tupelseite in 'T1 linker Outer Join T2', und T2 ist die Tupelseite in 'T1 rechter Outer Join T2'.

Kapitel 48. Leistungsaspekte bei der Suche

Bei der Suche in SQL, der Suche mit gespeicherten Prozeduren oder der Verwendung der Funktion NUMBEROFMATCHES bzw. SCORE ohne die Funktion CONTAINS können Sie durch die Berücksichtigung bestimmter Aspekte die Suchleistung verbessern.

Beachten Sie die folgenden Punkte zur Verbesserung der Leistung während der Suche:

- Bei der Suche innerhalb von SQL:
 - Wenn Sie eine Abnahme Systemleistung feststellen, können Sie mit der Anweisung `explain` den Verarbeitungsplan des DB2-Optimierungsprogramms überprüfen.
 - Durch die parametrische Suche kann der Zeitaufwand für Suchoperationen reduziert werden. Dies gilt insbesondere bei Verwendung weiterer Vergleichselemente, um die Menge der Ergebnisdaten einzuschränken.
 - Verwenden Sie das Schlüsselwort für die Ergebnisbegrenzung, wenn Sie nicht alle Ergebnisse benötigen.
- Bei der Suche mit der gespeicherten Prozedur:
 - Da Daten gemäß dem angegebenen Cachetabellenausdruck aus der Datenbank in den Speicher kopiert werden, müssen Sie sicherstellen, dass Ihre Workstation über ausreichend Speicherkapazität für diese Daten verfügt. Falls nicht ausreichend Speicher vorhanden ist, wird Auslagerungsspeicher verwendet, wodurch sich die Suchleistung verringert.
- Wenn Sie die Funktion NUMBEROFMATCHES oder die Funktion SCORE ohne die Funktion CONTAINS verwenden, kann die Leistung beeinträchtigt werden. Stellen Sie außerdem sicher, dass die Zeichenfolge in der Funktion CONTAINS genau mit der Zeichenfolge in der Funktion NUMBEROFMATCHES oder der Funktion SCORE übereinstimmt. Somit vermeiden Sie eine doppelte Verarbeitung.

Kapitel 49. Benutzerszenarios

Net Search Extender kann bei der Ausführung einer skalaren SQL-Suche, einer Suche mit gespeicherten Prozeduren oder einer Suche mit einer SQL-Tabellenwertfunktion verwendet werden.

Dieses Kapitel gibt Ihnen die Möglichkeit, sich mit Net Search Extender vertraut zu machen; arbeiten Sie hierfür die folgenden Beispiele durch:

Beispiel für skalare SQL-Suche

Dieses Befehlszeilenbeispiel demonstriert die verfügbaren Indexierungs- und Suchfunktionen.

Beispiel mit gespeicherter Prozedur

Dieses Befehlszeilenbeispiel arbeitet mit dem Indexierungsbefehl aus dem obigen Beispiel. Durch das Hinzufügen eines Caches demonstriert das Beispiel jedoch die unterschiedlichen für die Suche mit gespeicherten Prozeduren verfügbaren Indexierungs- und Suchfunktionen.

Beispiel für SQL-Tabellenwertfunktion

Das Beispiel für die SQL-Tabellenwertfunktion ist eine Variante für das Beispiel mit der Suche mit gespeicherten Prozeduren.

Anmerkung: Stellen Sie vor Verwendung der Beispiele mithilfe der Prozedur zur Installationsüberprüfung sicher, dass Net Search Extender erfolgreich installiert wurde.

Einfaches Beispiel mit der skalaren SQL-Suchfunktion

Sie können die im DB2 Net Search Extender-Beispiel für die skalare SQL-Suchfunktion beschriebenen Schritt ausführen.

Beispiel

Führen Sie folgende Schritte des DB2 Net Search Extender-Beispiels aus:

1. Erstellen einer Datenbank
2. Aktivieren einer Datenbank für die Textsuche
3. Erstellen einer Tabelle
4. Erstellen eines Volltextindex
5. Laden der Beispieldaten
6. Synchronisieren des Textindex
7. Ausführen von Suchoperationen mit dem Textindex

Sie können die Beispielbefehle in der Befehlszeile des Betriebssystems unter Verwendung einer vorhandenen Datenbank ausführen. Für die folgenden Beispiele lautet der Name der Datenbank `sample`.

Erstellen einer Datenbank

Sie können in DB2 mithilfe des folgenden Befehls eine Datenbank erstellen:
`db2 create database sample`

Aktivieren einer Datenbank für die Textsuche

Sie können DB2 Net Search Extender-Befehle auf die gleiche Weise wie

DB2-Befehle über die Befehlszeile des Betriebssystems ausführen. Verwenden Sie beispielsweise folgenden Befehl zum Starten von Net Search Extender-Instanzservices:

```
db2text START
```

Bereiten Sie anschließend die Datenbank für die Verwendung mit DB2 Net Search Extender vor:

```
db2text ENABLE DATABASE FOR TEXT CONNECT TO sample
```

Sie müssen diesen Schritt nur einmal für jede Datenbank ausführen.

Erstellen einer Tabelle

```
db2 "CREATE TABLE books (isbn VARCHAR(18) not null PRIMARY KEY,  
author VARCHAR(30), story CLOB(100k), year INTEGER)"
```

Mit diesem DB2-Befehl wird die Tabelle books erstellt. Diese enthält Spalten für author (Autor), story (Geschichte), isbn number (ISBN) und year (Erscheinungsjahr des Buches). Beachten Sie, dass die Tabelle über einen Primärschlüssel verfügen muss.

Erstellen eines Volltextindex

```
db2text "CREATE INDEX db2ext.myTextIndex FOR TEXT ON books (story)  
CONNECT TO sample"
```

Dieser Befehl erstellt einen Volltextindex für die Spalte story. Der Name des Textindex lautet db2ext.myTextIndex.

Laden von Beispieldaten

```
db2 "INSERT INTO books VALUES ('0-13-086755-1','John', 'A man was  
running down the street.',2001)"  
db2 "INSERT INTO books VALUES ('0-13-086755-2','Mike', 'The cat hunts  
some mice.', 2000)"  
db2 "INSERT INTO books VALUES ('0-13-086755-3','Peter', 'Some men  
were standing beside the table.',1999)"
```

Mit diesen Befehlen werden die Informationen für isbn, author, story und publishing year für drei Bücher in die Tabelle geladen.

Synchronisieren des Textindex

Geben Sie folgenden Befehl ein, um den Textindex mit den Daten aus der Tabelle sample zu aktualisieren:

```
db2text "UPDATE INDEX db2ext.myTextIndex FOR TEXT CONNECT TO sample"
```

Ausführen von Suchoperationen mit dem Textindex

Verwenden Sie zum Durchsuchen des Textindex die folgende skalare Suchfunktion CONTAINS:

```
db2 "SELECT author, story FROM books WHERE CONTAINS  
(story, '\"cat\"') = 1 AND YEAR >= 2000"
```

Anmerkung: Abhängig von der Betriebssystemshell, die Sie verwenden, müssen Sie möglicherweise unterschiedliche Escapezeichen vor den doppelten Anführungszeichen verwenden, die den Textsuchausdruck einschließen. Das oben dargestellte Beispiel verwendet als Escapezeichen "\".

Diese Abfrage sucht nach allen Büchern mit dem Begriff cat, wobei der Wert year des Buchs größer als oder gleich 2000 ist. Die Abfrage gibt die folgende Ergebnistabelle zurück:

```
AUTHOR Mike  
STORY The cat hunts some mice.
```

Weitere unterstützte Funktionen sind SCORE und NUMBEROFMATCHES. SCORE gibt einen Bezugswert darüber zurück, wie gut ein gefundenes Dokument durch den Suchbegriff beschrieben wird. NUMBEROFMATCHES gibt zurück, wie viele Übereinstimmungen mit den Suchbegriffen in jedem der gefundenen Dokumente enthalten sind.

Einfaches Beispiel mit Verwendung des Cache und der Suche mit gespeicherten Prozeduren

Sie können die im DB2 Net Search Extender-Beispiel für die Verwendung des Cache beschriebenen Schritte ausführen.

Beispiel

Führen Sie die folgenden Schritte im Beispiel für eine Suche mit gespeicherten Prozeduren in DB2 Suche mit gespeicherten Prozeduren aus:

1. Erstellen eines Textindex mit Cacheoption.
2. Synchronisieren des Index und Aktivieren des Cache.
3. Ausführen einer Suchoperation mit der gespeicherten Prozedur TEXTSEARCH.

Anmerkung: Das Beispiel für die Suche mit gespeicherten Prozeduren setzt voraus, dass die Schritte aus dem vorherigen Beispiel ausgeführt wurden und die Datenbank immer noch aktiviert ist.

Erstellen eines Textindex mit Cacheoption

Da die Datenbank bereits aktiviert ist, verwenden Sie den folgenden Befehl zum Erstellen eines Volltextindex:

```
db2text "CREATE INDEX db2ext.mySTPTextIndex FOR TEXT ON books (story)
        CACHE TABLE (author, story) MAXIMUM CACHE SIZE 1
        CONNECT TO sample"
```

In diesem Beispiel gilt der Volltextindex für die Spalte story; er gibt eine Cachetabelle mit den Spalten author und story an. Der Name des Textindex lautet mySTPTextIndex.

Synchronisieren des Index und Aktivieren des Cache

Verwenden Sie den folgenden Befehl, um den Index mit den in die Tabelle eingefügten Daten zu aktualisieren:

```
db2text "UPDATE INDEX db2ext.mySTPTextIndex FOR TEXT CONNECT TO sample"
```

Geben Sie zur Aktivierung des Cache den folgenden Befehl ein:

```
db2text "ACTIVATE CACHE FOR INDEX db2ext.mySTPTextIndex FOR TEXT
        CONNECT TO sample"
```

Hierdurch wird der Inhalt der Spalten author und story in den Cache geladen.

Ausführen einer Suchoperation mit der gespeicherten Prozedur TEXTSEARCH

Die gespeicherte Prozedur des DB2 Net Search Extender kann nur in bestimmten Fällen verwendet werden.

```
db2 "call db2ext.textSearch
    ('\"cat\"', 'DB2EXT', 'MYSTPTTEXTINDEX', 0,2,0,0,?,?)"
```

Diese Abfrage sucht nach allen Büchern zum Begriff cat (Katze), gibt jedoch nur die ersten beiden Ergebnisse zurück. Die Ergebnistabelle für ein Buch kann wie folgt aussehen:

Value of output parameters

```
-----  
Parameter Name : SEARCHTERMCOUNTS  
Parameter Value : 1  
Parameter Name : TOTALNUMBEROFRESULTS  
Parameter Value : 1
```

```
AUTHOR    STORY  
Mike      The cat hunts some mice.  
Return Status = 0
```

Weitere Beispiele zur Suchsyntax finden Sie in der folgenden Datei im DB2-Instanzverzeichnis: `sql1lib/samples/extenders/db2ext/search`.

Einfaches Beispiel mit der SQL-Tabellenwertfunktion

Sie können die SQL-Tabellenwertfunktion für die Textindizes verwenden, die in den vorherigen Beispielen erstellt wurden.

Informationen zu diesem Vorgang

Die Abfrage mit der SQL-Tabellenwertfunktion entspricht der zuvor verwendeten Abfrage mit der Funktion CONTAINS. Weitere Informationen hierzu finden Sie im Abschnitt zur Synchronisation des Textindex im Thema „Einfaches Beispiel mit der skalaren SQL-Suchfunktion“ auf Seite 157.

```
db2 "SELECT author, story FROM books b, table (db2ext.textsearch  
      ('\"cat\"','DB2EXT','MYTEXTINDEX', 0, 2, CAST  
      (NULL AS VARCHAR(18)))) T where T.primKey = b.isbn
```

In diesem Beispiel wird NULL in den Datentyp des Primärschlüssels umgesetzt.

Kapitel 50. Verwenden eines Thesaurus zur Erweiterung von Suchbegriffen

Sie können eine Abfrage weiter fassen, indem Sie nicht nur nach einem bestimmten Suchbegriff suchen, sondern auch nach Begriffen, die zu diesem Begriff in einer Beziehung stehen. Sie können diesen Prozess automatisieren, indem Sie die Funktionen von Net Search Extender zum Suchen und Extrahieren der zugehörigen Suchbegriffe aus einem Thesaurus verwenden.

Ein Thesaurus ist ein abgegrenzter Wortschatz semantisch zusammengehöriger Begriffe, die sich in der Regel auf ein bestimmtes Themengebiet beziehen.

Net Search Extender gibt Ihnen die Möglichkeit, einen Suchbegriff durch Hinzufügen zusätzlicher Begriffe aus einem Thesaurus zu erweitern, den Sie zuvor erstellt haben. Informationen zur Verwendung der Thesauruserweiterung in einer Abfrage finden Sie in Kapitel 59, „Syntax für Suchargumente“, auf Seite 257.

Die Erstellung eines Thesaurus zur Verwendung in einer Suchanwendung erfordert eine Thesaurusdefinitionsdatei, die in ein internes Format, das Thesauruswörterverzeichnis, kompiliert werden muss.

In diesem Abschnitt wird Folgendes behandelt:

- **„Struktur eines Thesaurus“**

Ein Thesaurus ist wie ein Netz aus Knoten strukturiert, die durch Relationen miteinander verknüpft sind. Dieser Abschnitt beschreibt die vordefinierten Relationen von Net Search Extender und erläutert, wie Sie eigene Relationen definieren können.

- **„Erstellen und Kompilieren eines Thesaurus“ auf Seite 163**

Dieser Abschnitt enthält eine Beschreibung der Syntax einer Thesaurusdefinitionsdatei sowie der Tools, die Sie verwenden, um sie in ein Thesauruswörterverzeichnis zu kompilieren.

Struktur eines Thesaurus

Ein Thesaurus ist wie ein Netz aus Knoten strukturiert, die durch Relationen miteinander verknüpft sind.

Net Search Extender sucht einen Begriff in einem Thesaurus, indem er bei dem Begriff beginnt, einem Pfad durch die zugehörigen Begriffsrelationen folgt und die während dieses Vorgangs gefundenen Begriffe zurückgibt.

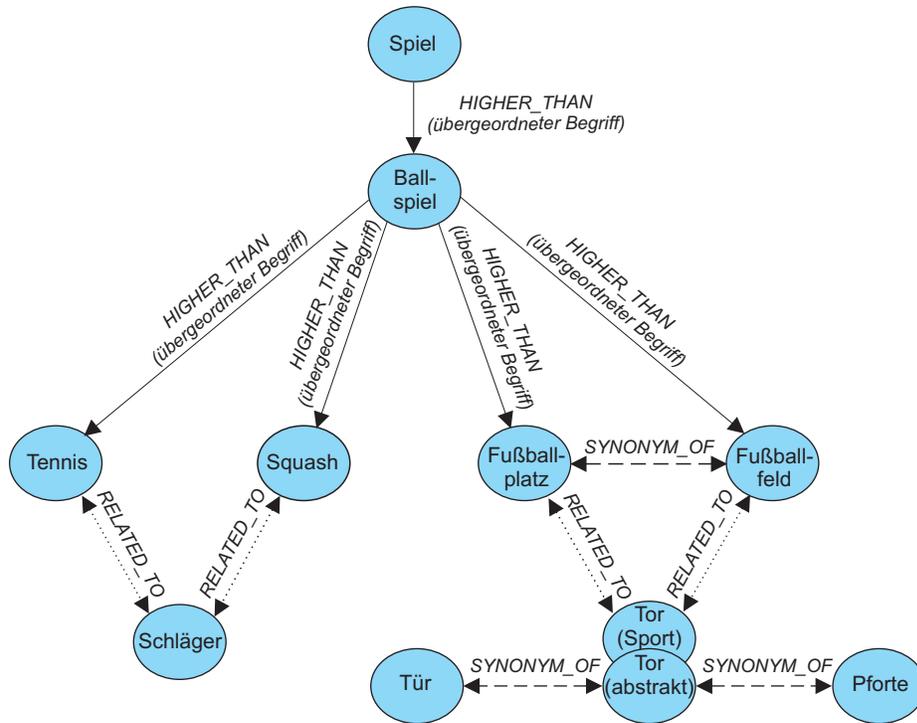


Abbildung 7. Beispiel für die Struktur eines Thesaurus

Thesauruseinträge sind durch Relationen verknüpft. Relationsnamen wie z. B. BROADER ermöglichen Ihnen, eine Erweiterung auf bestimmte benannte Linien in der Relationshierarchie zu beschränken. Einige Relationen funktionieren in zwei Richtungen (bidirektionale Relationen), andere jedoch nur in einer Richtung (unidirektionale Relationen). BROADER ist zum Beispiel der Name einer unidirektionalen Relation.

Vordefinierte Thesaurusrelationen

Net Search Extender enthält vordefinierte Relationen, wie zum Beispiel assoziative Relationen, Synonymrelationen und hierarchische Relationen.

Die folgenden Relationen sind in Net Search Extender vordefiniert:

- **Assoziative Relationen**

Eine assoziative Relation ist eine gegenseitige (bidirektionale) Zuordnung zwischen zwei Begriffen, die nicht dasselbe Konzept ausdrücken, jedoch zueinander in Beziehung stehen.

Vordefinierte assoziative Relation: RELATED_TO

Beispiele:

Tennis RELATED_TO Schläger
Fußball RELATED_TO Tor (Sport)

- **Synonymrelationen**

Eine Synonymrelation ist eine gegenseitige (bidirektionale) Zuordnung zwischen zwei Begriffen, die dieselbe oder ähnliche Bedeutung haben und als alternative Ausdrucksweisen verwendet werden können. Diese Relation kann zum Beispiel zwischen einem Begriff und seiner Abkürzung verwendet werden.

Vordefinierte Synonymrelation: SYNONYM_OF

Beispiele:

Position SYNONYM_OF Stelle
USA SYNONYM_OF Vereinigte Staaten

Die Abbildung in Abb. 7 auf Seite 162 zeigt zwei Begriffe Tor im gleichen Thesaurus. Der eine ist mit dem Kommentar (Sport), der andere mit dem Kommentar (abstrakt) angegeben. Auch wenn Begriffe die gleiche Schreibweise haben, können Synonymrelationen verschiedene Wortgruppen verknüpfen. Sie können dies modellieren, indem Sie bei der Definition verschiedene Relationen verwenden.

- **Hierarchische Relationen**

Eine hierarchische Relation ist eine Relation in einer Richtung (unidirektional) zwischen zwei Begriffen, von denen einer eine weiter gefasste (globalere) Bedeutung als der andere hat. Abhängig von der Richtung kann die Relation zur Suche nach einem spezielleren oder einem allgemeineren Begriff verwendet werden.

Vordefinierte hierarchische Relationen:

- LOWER_THAN zur Modellierung bedeutungsverengender Relationen
LOWER_THAN-Relationen dienen zur Modellierung einer Folge angepassterer Begriffe. Je tiefer der bedeutungsverengenden Relation gefolgt wird, desto spezieller werden die Begriffe. Wenn Sie zum Beispiel nach dem Begriff Ballspiel in einer LOWER_THAN-Relation suchen, könnte das Ergebnis Begriffe wie Squash usw. in einer Liste zunehmend speziellerer Begriffe liefern.
- HIGHER_THAN zur Modellierung bedeutungserweiternder Relationen
HIGHER_THAN-Relationen dienen zur Modellierung einer Folge immer allgemeinerer Begriffe. Je tiefer einer solchen Relation gefolgt wird, desto weniger spezifisch werden die Begriffe. Wenn Sie zum Beispiel nach dem Begriff Ballspiel in einer HIGHER_THAN-Relation suchen, könnte das Ergebnis Begriffe wie Spiel usw. in einer Liste zunehmend allgemeinerer Begriffe liefern.

Definieren eigener Relationen

Net Search Extender gibt Ihnen die Möglichkeit, eigene Thesaurusrelationen der Typen RELATED_TO, LOWER_THAN und HIGHER_THAN zu definieren.

Da jeder Relationsname eindeutig sein muss, müssen Sie solche Relationsnamen durch Hinzufügen einer eindeutigen Nummer zum Beispiel wie folgt qualifizieren: RELATED_TO(42).

Sie können die gleiche Relationsnummer zur Definition einer Beziehung eines anderen Typs verwenden, z. B. LOWER_THAN(42). Die Nummer 0 dient zum Verweis auf vordefinierte Relationen von Net Search Extender.

Erstellen und Kompilieren eines Thesaurus

Bei der Erstellung eines Thesaurus, der von den Net Search Extender-Funktionen genutzt werden kann, muss eine Reihe von Schritten ausgeführt werden.

Führen Sie folgende Schritte aus, um einen Thesaurus zu erstellen, der von den Net Search Extender-Funktionen verwendet werden kann:

1. Erstellen Sie eine Thesaurusdefinitionsdatei.
2. Kompilieren Sie die Definitionsdatei in ein Thesauruswörterverzeichnis.

Erstellen einer Thesaurusdefinitionsdatei

Bei der Erstellung einer Thesaurusdefinitionsdatei gelten bestimmte Einschränkungen.

Informationen zu diesem Vorgang

Ihr erster Schritt zur Erstellung eines eigenen Thesaurus besteht darin, seinen Inhalt in einer Definitionsdatei mithilfe eines Texteditors zu definieren.

Einschränkungen. Die Länge des Dateinamens, einschließlich der Erweiterung, darf 256 Zeichen nicht überschreiten. Sie können mehrere Thesauren im gleichen Verzeichnis speichern, jedoch wird empfohlen, für jeden Thesaurus ein separates Verzeichnis anzulegen.

Eine englische Beispieldefinitionsdatei für einen Thesaurus `nseampl.thes.def` wird bereitgestellt. Das Thesaurusverzeichnis für Windows-Systeme lautet wie folgt:

```
sqllib\db2ext\thes
```

Auf UNIX-Systemen lautet das Thesaurusverzeichnis wie folgt:

```
ausgangsverzeichnis_des_instanzeigners/sqllib/db2ext/thes
```

Das folgende Beispiel zeigt die ersten Definitionsgruppen aus dieser Datei:

```
:WORDS
    accounting
    .RELATED_TO account checking
    .RELATED_TO sale management
    .SYNONYM_OF account
    .SYNONYM_OF accountant

:WORDS
    acoustics
    .RELATED_TO signal processing

:WORDS
    aeronautical equipment
    .SYNONYM_OF turbocharger
    .SYNONYM_OF undercarriage

:WORDS
    advertising
    .RELATED_TO sale promotion
    .SYNONYM_OF advertisement
:
:
:
```

Abbildung 8. Ein Auszug auf der Beispieldefinitionsdatei für einen Thesaurus

Informationen zur Syntax der einzelnen Definitionsgruppen finden Sie in „Thesaurusunterstützung“ auf Seite 165.

Jeder Begriffseintrag muss in eine einzige Zeile geschrieben werden. Jedem zugeordneten Begriff muss ein Relationsname vorangestellt werden. Wenn die Begriffseinträge eine Beziehung zueinander haben, geben Sie eine Begriffsrelation an.

Die Länge der Begriffseinträge und der zugeordneten Begriffe ist auf 64 Zeichen begrenzt. Einzelbyteversionen und Doppelbyteversionen des gleichen Buchstabens werden als gleich betrachtet. Groß- und Kleinbuchstaben werden nicht unterschieden. Ein Term kann ein Leerzeichen enthalten. Außerdem ist entweder das Einzelbytezeichen Punkt (.) oder Doppelpunkt (:) zulässig.

Die benutzerdefinierten Relationen basieren alle auf dem *assoziativen* Typ. Sie werden durch eindeutige Nummern zwischen 1 und 128 gekennzeichnet.

Kompilieren einer Definitionsdatei in ein Thesauruswörterverzeichnis

Zum Kompilieren einer Thesaurusdefinitionsdatei führen Sie den Befehl **db2extth** aus.

Informationen zu diesem Vorgang

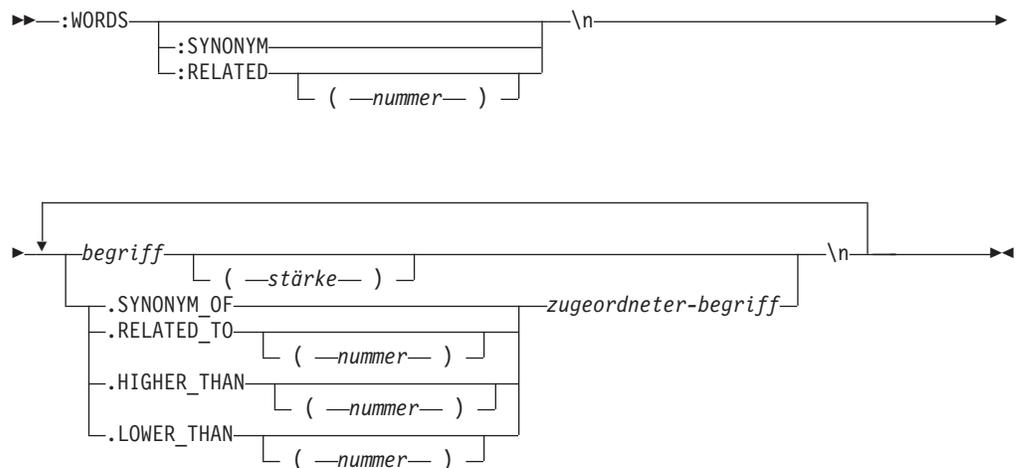
Zur Verwendung eines Thesauruswörterverzeichnisses in einer partitionierten Umgebung müssen Sie sicherstellen, dass alle physischen Knoten auf die erstellten Dateien zugreifen können.

Thesaurusunterstützung

Bei der Erstellung eines eigenen Thesaurus müssen Sie eine bestimmte Syntax verwenden.

Die Syntax der einzelnen Definitionsgruppen bei der Erstellung Ihres eigenen Thesaurus sieht folgendermaßen aus:

Syntax einer Thesaurusdefinition



Beachten Sie, dass \n nicht Bestandteil der Syntax ist, sondern das Ende einer Zeile in der Thesaurusdefinitionsdatei darstellt.

Sie können wie folgt Kommentarzeilen in eine Thesaurusdefinitionsdatei einfügen:

mein Kommentartext

:WORDS

Ein Schlüsselwort, das den Anfang einer Gruppe zusammengehöriger Wörter markiert.

:SYNONYM, :RELATED [(nummer)],

Ein Relationsname.

Relationsnamen bestehen aus einem Relationstyp und einer Nummer.

Wenn die Nummer nicht angegeben wird, wird null angenommen. Dies ist

der vom System bereitgestellte Relationsname. :SYNONYM ist immer der vom System bereitgestellte Relationsname.

Relationsnamen, die mit einem Doppelpunkt beginnen, zum Beispiel :SYNONYM, leiten eine Liste von Wörtern ein, deren Zusammengehörigkeit durch dieselbe Relation definiert wird. Beispiel:

```
:WORDS
  :SYNONYM
    Stewardess
    Mitglied der Kabinencrew
    Flugbegleiter
```

- begriff* Ein Begriff, der in das Thesauruswörterverzeichnis eingefügt werden soll.
- Maximale Länge ist 64 Byte (42 Byte für Codepage UTF-8).
 - Einzelbyteversionen und Doppelbyteversionen des gleichen Buchstabens werden als gleich betrachtet.
 - Groß- und Kleinbuchstaben werden nicht unterschieden.
 - Ein Begriff kann ein Leerzeichen enthalten.
 - Das Einzelbytezeichen Punkt ('.') bzw. Doppelpunkt (':') kann nicht verwendet werden.

Dieser Parameter kann nützlich sein, wenn Sie nicht wollen, dass eine Thesaurussuche Wörter mit einschließt, die eine schwache Relation zum gesuchten Begriff haben. Die Stärke ist ein numerischer Wert aus dem Bereich von 1 bis 100. Der Standardwert ist 100.

.SYNONYM_OF, .RELATED_TO [(nummer)], .HIGHER_THAN [(nummer)], .LOWER_THAN [(nummer)]

Ein Relationsname. Der Relationsname .HIGHER_THAN entspricht der Abfragerelation BROADER, und .LOWER_THAN entspricht der Abfragerelation NARROWER. Relationsnamen bestehen aus einem Relationstyp und einer Nummer. Wenn die Nummer nicht angegeben wird, wird null angenommen. Dies ist der vom System bereitgestellte Relationsname. Der Relationsname .SYNONYM ist immer der vom System bereitgestellte Relationsname.

Relationsnamen, die mit einem Punkt beginnen, zum Beispiel .SYNONYM_OF, definieren die Relation zwischen einem Begriff und einem anderen. Beispiel:

```
:WORDS
  Stewardess
  .SYNONYM_OF Mitglied der Kabinencrew
  .SYNONYM_OF Flugbegleiter
```

Die optionale *nummer* identifiziert eine benutzerdefinierte Relation. Diese Nummer muss ein eindeutiger Wert aus der Thesaurusdefinitionsdatei sein (derzeit 1 bis 128). Beispiel: RELATED_TO(42).

Wenn Sie symbolische Namen für Thesaurusrelationen in Ihrer Anwendung anstelle des Relationsnamens und der Nummer verwenden wollen, muss Ihre Anwendung die Zuordnung von Name zu Nummer übernehmen. Wenn Sie zum Beispiel die Relation GEGENTEIL_VON als RELATED_TO(1) definieren, muss Ihre Anwendung diesen Namen dem internen Relationsnamen RELATED_TO(1) zuordnen.

zugeordneter-begriff

Jedem zugeordneten Begriff muss ein Relationsname vorangestellt werden. Der zugeordnete Begriff wird jedem Begriff in Bezug auf die angegebene Relation zugeordnet. Wenn alle Begriffe eine Relation zueinander haben, kann dies mithilfe einer Begriffsrelation angegeben werden.

- Maximale Länge ist 64 Byte (42 Byte für Codepage UTF-8).

- Einzelbyteversionen und Doppelbyteversionen des gleichen Buchstabens werden als gleich betrachtet.
- Groß- und Kleinbuchstaben werden nicht unterschieden.
- Ein Begriff kann ein Leerzeichen enthalten.
- Das Einzelbytezeichen Punkt ('.') bzw. Doppelpunkt (':') kann nicht verwendet werden.

Im Folgenden sehen Sie ein Beispiel für einen zugeordneten Begriff:

```
:WORDS:SYNONYM
zurückweisen
ablehnen
RELATED_T0(1) akzeptieren
```

Vom Thesaurus unterstützte CCSIDs

Bestimmte CCSIDs werden durch den Thesaurus unterstützt.

Die folgenden IDs für codierte Zeichensätze (CCSIDs) werden durch den Thesaurus unterstützt:

819	Lateinischer Zeichensatz 1
850	Lateinischer Zeichensatz 1 (PC-Daten)
874	Thailändisch
932	Kombiniertes Japanisch
943	Kombiniertes Japanisch
949	Kombiniertes Koreanisch
950	Kombiniertes traditionelles Chinesisch
954	Japanisch
970	Kombiniertes Koreanisch
1208	UTF 8
1250	Lateinischer Zeichensatz 2
1252	Lateinischer Zeichensatz 1
1253	Tschechisch
1254	Türkisch
1255	Hebräisch
1256	Arabisch
1258	Vietnamesisch
1363	Kombiniertes Koreanisch
1381	Kombiniertes vereinfachtes Chinesisch
1383	Chinesisch (vereinfacht), SBCS/DBCS kombiniert
1386	Chinesisch (vereinfacht), SBCS/DBCS kombiniert
5039	Japanisch (SBCS/DBCS kombiniert)

Nachrichten des Thesaurustools

Bei der Verwendung des Thesaurustools kann eine Reihe von Fehlermeldungen zurückgegeben werden.

ADM_MSG_INVALID_CCSID

Es wurde eine ungültige ID für codierten Zeichensatz angegeben.

Die angeforderte Codepage wird nicht unterstützt.

ITL_THES_MSG_BUFFER_OVERFLOW

Pufferüberlauf.

ITL_THES_MSG_DICT_EXIST

Das Wörterverzeichnis *wörterverzeichnisname* des Thesaurus ist bereits vorhanden.

Es kann nicht überschrieben werden.

ITL_THES_MSG_DICT_INTEGRITY_ERROR

Integrität von Wörterverzeichnis *wörterverzeichnisname* ist verloren gegangen.

Die Wörterverzeichnisdatei des Thesaurus ist beschädigt.

ITL_THES_MSG_DICT_NOT_EXIST

Wörterverzeichnis *wörterverzeichnisname* des Thesaurus ist nicht vorhanden.

ITL_THES_MSG_DICT_VERSION_ERROR

Versionsfehler von Wörterverzeichnis *wörterverzeichnisname*.

Das Wörterverzeichnis des Thesaurus wurde mit einer inkompatiblen früheren Version erstellt.

ITL_THES_MSG_ERROR_IN_FILE

Fehler in Datei *dateiname*.

ITL_THES_MSG_FILE_ACCESS_ERROR

Auf Datei *dateiname* konnte nicht zugegriffen werden.

ITL_THES_MSG_FILE_CLOSE_ERROR

Datei *dateiname* konnte nicht geschlossen werden.

ITL_THES_MSG_FILE_EOF_ERROR

Unerwartetes Dateiende in *dateiname*.

Fehler in Definitionsdatei.

ITL_THES_MSG_FILE_OPEN_ERROR

Datei *dateiname* konnte nicht geöffnet werden.

ITL_THES_MSG_FILE_REACHED_END

Unerwartetes Dateiende in *thesaurusdefinitionsdatei*.

Die Definitionsdatei enthält einen Fehler.

ITL_THES_MSG_FILE_READ_ERROR

Datei *dateiname* konnte nicht gelesen werden.

ITL_THES_MSG_FILE_REMOVE_ERROR

Datei *dateiname* konnte nicht entfernt werden.

ITL_THES_MSG_FILE_RENAME_ERROR

Datei *dateiname 1* konnte nicht in *dateiname 2* umbenannt werden.

ITL_THES_MSG_FILE_WRITE_ERROR

In Datei *dateiname* konnte nicht geschrieben werden.

ITL_THES_MSG_IE_BLOCK_START

In Datei *dateiname* in Zeile *zeilennummer* wurde keine Blockanfangszeile gefunden.

ITL_THES_MSG_IE_EMPTY

Die Thesaurusdefinitionsdatei *dateiname* ist leer.

ITL_THES_MSG_IE_NO_TERM

In Datei *dateiname* in Zeile *zeilennummer* sind keine Begriffe definiert.

ITL_THES_MSG_IE_REL_SYNTAX

Relation in Datei *dateiname* in Zeile *zeilennummer* wurde falsch angegeben.

ITL_THES_MSG_IE_STRENGTH_DOMAIN

Die Stärke ist außerhalb des gültigen Bereichs.

Gültige Werte sind 1 - 100; Standardwert ist 100.

ITL_THES_MSG_IE_STRENGTH_SYNTAX

Ein Stärkewert ist falsch angegeben.

Syntax: Geben Sie nach dem Begriff [:20] für die Stärke 20 ein.

ITL_THES_MSG_IE_TERM_LEN

Ein Thesaurusbegriff ist länger als 64 Zeichen.

ITL_THES_MSG_IE_USER_DEF

Relation in Datei *dateiname* in Zeile *zeilennummer* wurde falsch angegeben.

ITL_THES_MSG_IE_USER_DEF_DOMAIN

Relationsnummer in Datei *dateiname* in Zeile *zeilennummer* ist außerhalb des gültigen Bereichs.

ITL_THES_MSG_INPUT_ERROR

Fehler in der Thesaurusdefinitionsdatei *dateiname* in Zeile *zeilennummer*.

ITL_THES_MSG_INTERNAL_ERROR

Interner Fehler.

ITL_THES_MSG_LOCKED

Thesauruswörterverzeichnis *wörterverzeichnisname* wird verwendet.

ITL_THES_MSG_LOCKING_ERROR

Wörterverzeichnis *dateiname* konnte nicht gesperrt werden.

ITL_THES_MSG_MEMORY_ERROR

Speicherfehler.

ITL_THES_MSG_NAMELEN_ERROR

Parameterfehler: *dateiname*. Der Name der Thesaurusdefinitionsdatei ist zu lang.

ITL_THES_MSG_NO_TARGET_DIR_ERROR

Parameterfehler. Es wurde kein Zielverzeichnis angegeben.

ITL_THES_MSG_NONAME_ERROR

Parameterfehler. Es wurde keine Thesaurusdefinitionsdatei angegeben.

ITL_THES_MSG_NORMALIZE_ERROR

Fehler beim Normalisieren eines Begriffs.

Fehler in der Thesaurusdefinitionsdatei.

ITL_THES_MSG_OUTFILE_EXIST

Ausgabedatei *dateiname* ist bereits vorhanden.

ITL_THES_MSG_PARAMETER_ERROR

Interner Parameterfehler.

ITL_THES_MSG_PATHLEN_ERROR

Parameterfehler: *dateiname*. Der Pfad der Thesaurusdefinitionsdatei ist zu lang. Die Pfadlänge darf nicht die Maximallänge für Verzeichnisnamen überschreiten, die vom Betriebssystem unterstützt wird.

ITL_THES_MSG_UNEXPECTED_ERROR

Unerwarteter interner Fehler.

Kapitel 51. Net Search Extender-Indexierungskonfiguration

Es stehen bestimmte Konfigurationsoptionen zur Verfügung, mit denen das Indexierungs- und Suchverhalten von Net Search Extender geändert werden kann.

In diesem Kapitel finden Sie Informationen zu einigen Konfigurationsoptionen, mit denen das Indexierungs- und Suchverhalten von Net Search Extender geändert werden kann.

- Tokenanalyse
- Stoppwörter
- Konfiguration

Tokenanalyse

Bei der Indexierung verarbeitet Net Search Extender einen Dokumenttext wie folgt, wobei der Text in Token (lexikalische Einheiten) zerlegt wird.

Wörter

Alle alphanumerischen Zeichen ('a'..'z','A'..'Z', '0'..'9') werden zur Erstellung des Volltextindex verwendet. Trennzeichen sind Leerzeichen und die Zeichen, die im Abschnitt zur Satzerkennung weiter unten beschrieben werden. Steuerzeichen, wie das Zeilenvorschubzeichen (auch als Zeilenschaltung bezeichnet) und Leerzeichen, werden folgendermaßen interpretiert: Steuerzeichen (mit kleinerer Nummer als 0x20) in der Mitte der Zeile werden als Leerzeichen betrachtet. Leerzeichen und Steuerzeichen vor und nach einem Zeilenvorschub (0x0A) werden ignoriert. Ein Zeilenvorschubzeichen vor und nach einem 1-Byte-Zeichen werden als Leerzeichen und 2-Byte-Zeichen für das gleiche Zeichen werden immer als das gleiche Zeichen betrachtet. Die große Form und die kleine Form des gleichen Buchstabens, zum Beispiel 'A' und 'a', werden als gleiche Zeichen betrachtet, wenn beim Suchen nichts angegeben wird, oder als unterschiedliche Zeichen, wenn beim Suchen exakte Übereinstimmung angefordert wird.

Sätze

Net Search Extender erkennt die Zeichen '.', '!', '?', vorausgesetzt, die folgenden Bedingungen sind erfüllt:

- Auf das Sonderzeichen '.' muss ein Leerzeichen oder ein Zeilenumbruch folgen, damit es als Satzende interpretiert wird.
- Die Zeichen '!' und '?' an einer beliebigen Stelle im Text werden als Satzende interpretiert (auch ohne nachfolgendes Leerzeichen oder nachfolgenden Zeilenumbruch).
- Wenn '!', '?' oder '.' in Anführungszeichen stehen, werden sie ignoriert und nicht als Satzende interpretiert.

Absätze

Die Erkennung von Absätzen hängt vom Dokumentformat ab. Im reinen Textformat werden jeweils zwei aufeinander folgende Zeilenvorschubzeichen (ggf. mit einem dazwischen stehenden Wagenrücklaufzeichen) als Absatzgrenze erkannt. Im HTML-Format wird der Absatzformatierungsbefehl <p> als Absatzgrenze interpretiert. Für die anderen Dokumentformate wird keine Absatzerkennung unterstützt.

Stoppwörter

Stoppwörter sind Wörter, die häufig vorkommen und für den Textabrufprozess keinen relevanten Inhalt besitzen.

In der Regel werden alle Funktionswörter (in linguistischem Sinn) als Stoppwörter betrachtet, zum Beispiel „und“, „oder“ und „in“. Das Durchsuchen eines Index nach Stoppwörtern kann die Genauigkeit eines Textabrufsystems beachtlich reduzieren.

Net Search Extender stellt die Stoppwörterverarbeitung für eine Reihe von Sprachen bereit. Der Konfigurationsparameter **IndexStopWords** kann bei der Indexerstellung gesetzt werden; er legt fest, ob Stoppwörter indiziert werden. Der Standardwert ist 1; dies bedeutet, dass Stoppwörter indiziert werden.

Wenn Sie keine Stoppwörter indizieren möchten, müssen Sie den Parameter **IndexStopWords** auf 0 setzen und die Sprache Ihrer Eingabedokumente mit dem Sprachenparameter bei der Indexerstellung angeben. Wenn keine Stoppwörter indiziert werden, ist der Index kleiner und schneller. Ändern Sie diesen Wert nicht in der Konfigurationsdateischablone mit der Endung `.ini`, wenn Sie den Index erstellt haben, da dies dazu führt, dass Dokumente in Abhängigkeit von der Zeit, zu der sie indiziert wurden, unterschiedlich behandelt werden und dass folglich eine inkohärente Stoppwörterbehandlung daraus resultiert.

Das Ignorieren von Stoppwörtern beim Indexieren ist nur wirksam, wenn alle Dokumente in Ihrer Sammlung in derselben Sprache vorhanden sind.

Sprachen mit Stoppwörterunterstützung

Für bestimmte Sprachen wird eine Stoppwörterverarbeitung unterstützt.

Für die folgenden Sprachen wird eine Stoppwörterverarbeitung unterstützt.

AR_AA

Arabisch, wie in arabischen Ländern gesprochen

CA_ES

Katalanisch, wie in Spanien gesprochen

DA_DK

Dänisch, wie in Dänemark gesprochen

DE_CH

Deutsch, wie in der Schweiz gesprochen

DE_DE

Deutsch, wie in Deutschland gesprochen

EL_GR

Griechisch, wie in Griechenland gesprochen

EN_GB

Englisch, wie in GB gesprochen

EN_US

Englisch, wie in den USA gesprochen

ES_ES Spanisch, wie in Spanien gesprochen

FI_FI Finnisch, wie in Finnland gesprochen

FR_CA	Französisch, wie in Kanada gesprochen
FR_FR	Französisch, wie in Frankreich gesprochen
HE_IL	Hebräisch, wie in Israel gesprochen
IS_IS	Isländisch, wie auf Island gesprochen
IT_IT	Italienisch, wie in Italien gesprochen
IW_IL	Hebräisch, wie in Israel gesprochen
NB_NO	Norwegisches Bokmål, wie in Norwegen gesprochen
NL_BE	Niederländisch, wie in Belgien gesprochen
NN_NO	Norwegisches Nynorsk, wie in Norwegen gesprochen
PT_BR	Portugiesisch, wie in Brasilien gesprochen
PT_PT	Portugiesisch, wie in Portugal gesprochen
RU_RU	Russisch, wie in Russland gesprochen
SV_SE	Schwedisch, wie in Schweden gesprochen

Konfiguration

Net Search Extender kann nach Wörtern mit Zeichen in verschiedenen Kombinationen suchen, z. B. alphanumerische Zeichen, Zahlen und Sonderzeichen.

Hierfür stellt Net Search Extender die folgenden Konfigurationen zur Verfügung:

Zeichennormalisierung

Mit der Zeichennormalisierung wird sichergestellt, dass nach Wörtern gesucht werden kann, für die es zwei Schreibweisen gibt. So kann z. B. das Wort 'Überbau' auch mit 'ue' geschrieben werden. Mit der Normalisierung wird sichergestellt, dass nach beiden Wörtern gesucht werden kann, also sowohl nach 'Überbau' als auch nach 'Ueberbau'. Mit der Funktion werden auch Buchstaben mit Akzent, z. B. 'accès' für das übereinstimmende Einzelzeichen normalisiert (z. B. 'acces'). Beachten Sie, dass die Verwendung dieser Option zu unerwünschten Ergebnissen in Sprachen führen kann, in denen z. B. das Zeichen 'Ü' keine funktional entsprechende Normalisierung als 'Ue' aufweist.

Sonderzeichen als Teil eines Worts verwenden

Mit der Verwendung von Sonderzeichen in einem Wort stellen Sie sicher, dass nach Produktnamen, in denen oftmals eine Kombination von alphanumerischen Zeichen, Sonderzeichen und Zahlen verwendet wird, als Wort gesucht werden kann. Beispiel: Wenn Sie die alphanumerische Kombination 'DT9' als ein Wort betrachten oder das Sonderzeichen '/' aktivieren, wird z. B. nach OS/390 als Ganzes gesucht und nicht nach 'OS' und '390'.

Für diese Konfigurationseinstellungen sind Schalter verfügbar. Um die Schalter anzupassen, ändern Sie vor der Erstellung eines Index die Dateischablone `.ini`.

Die Dateischablone `.ini` befindet sich im Verzeichnis `sql11lib/db2ext/cteixcfg.ini`. Da Sie mit dem Befehl **CREATE INDEX** auch Änderungen an den meisten Werten dieser Schablonendatei vornehmen können, sollten Sie nur Änderungen an den folgenden Werten vornehmen:

`AccentRemoval` (zur Zeichennormalisierung)

`UmlautNormalization` (zur Zeichennormalisierung)

`TreatNumberAsWords` (damit numerische Zeichen als Teil des Worts behandelt werden)

`AdditionalAlphanumCharacters` (damit bestimmte Zeichen als Teil eines Worts verwendet werden)

AccentRemoval

Dieser Parameter gibt an, ob Zeichen mit Akzent in das übereinstimmende Einzelzeichen normalisiert werden. Beispiel: 'événement' wird auch als 'evenement' indiziert. Der Standardwert ist 'true'.

UmlautNormalization

Dieser Parameter gibt an, ob ein Umlautzeichen auch als zwei Zeichen mit derselben Bedeutung indiziert wird. Beispiel: 'Übersee' wird auch als 'Uebersee' indiziert. Der Standardwert ist 'true'.

TreatNumbersAsWords

Dieser Parameter gibt an, ob numerische Zeichen neben einem Wort Teil des Worts sind. Beispiel: 'DT9' wird als ein Wort und nicht als Wort 'DT' und Zahl '9' behandelt.

AdditionalAlphanumCharacters

Der Zeichenfolgewart dieses Parameters definiert, welche Zeichen als Teil eines Worts behandelt werden. Die Zeichenfolge von Sonderzeichen muss eine Folge von mindestens einem Zeichen in UTF-8 sein. Die Standardzeichenfolge enthält die Zeichen `'/-@'`.

Die Verwendung der Platzhalterzeichen `%` und `_` in der Liste der Zeichen, die als Teil eines Wortes behandelt werden, ist nicht zulässig. Dies führt zu Problemen während der Abfrageausführung.

Wenn Sie diese Konfigurationswerte ändern möchten, bearbeiten Sie die Datei mit der Endung `.ini`, bevor Sie Ihren Index erstellen. Um inaktive Schalter zu aktivieren, entfernen Sie vor jeder Zeile die Kommentarmarke `'`. Weitere Informationen finden Sie in der Datei `cteixcfg.ini`.

Es ist empfehlenswert, keinen anderen Wert in der Datei mit der Endung `.ini` zu ändern.

Teil 12. Arbeiten mit strukturierten Dokumenten

Net Search Extender gibt Ihnen die Möglichkeit, Textfelder oder numerische Felder zu indexieren und zu durchsuchen, wie zum Beispiel Felder für Titel (title), Autor (author) oder Preis (price) in einem strukturierten Dokument.

Die Dokumente können im XML-, Outside In- oder HTML-Format vorliegen oder benutzerdefinierte Tags (GPP-Format) enthalten.

Verwenden Sie Formatierungssteuerbefehle und die zugehörigen Feldnamen in einem *Dokumentmodell*, um zu definieren, welche Felder in den Dokumenten indexiert werden sollen und deshalb für die Suche verfügbar sein müssen. Der Name des Feldes (auch als Abschnittsname bezeichnet) kann in Abfragen auf dieses Feld verwendet werden.

Zum Durchsuchen dieser Felder müssen Sie eine Format- und Modelldatei (FORMAT und MODEL) angeben, wenn Sie den Textindex erstellen, der die Dokumente enthält.

Kapitel 52. Suche in nativ gespeicherten XML-Dokumenten

Im Allgemeinen müssen Sie beim Erstellen eines Index in einer XML-Datenspalte kein FORMAT angeben. Net Search Extender wählt standardmäßig das XML-Format aus, wenn in einer Spalte vom Typ XML ein Textindex erstellt wird.

Die folgenden Abschnitte betreffen die Suche in nativ gespeicherten XML-Dokumenten. Sie sehen, wie die Konzepte der Abschnittssuche auf nativ gespeicherte XML-Dokumente angewendet werden können und wie diese Funktionalität in die XQuery-Verarbeitung integriert werden kann.

Im Allgemeinen müssen Sie beim Erstellen eines Index in einer XML-Datenspalte kein FORMAT angeben. Net Search Extender wählt standardmäßig das XML-Format aus, wenn in einer Spalte vom Typ XML ein Textindex erstellt wird. Die Formatkennungen TEXT und HTML sind in XML-Datenspalten nicht zulässig.

In den folgenden Beispielen, bei denen die Erstellung und Verwendung eines Textindex in XML-Spalten dargestellt wird, wird das folgende XML-Dokument verwendet. Die Speicherung findet in Tabelle t1, Spalte c2 vom Typ XML statt.

```
<?xml version="1.0">
<purchaseOrder orderDate="2001-01-20">
  <shipAddress countryCode="US">
    <name>Alice Smith</name>
    <street>123 Maple Street</street>
    <city>Mill Hill</city>
    <zip>90999</zip>
  </shipAddress>
  <item partNo="123" quantity="1">
    <name>S&B Lawnmower Type ABC-x</name>
    <price>239.90</price>
    <shipDate>2001-01-25</shipdate>
  </item>
  <item partNo="987" quantity="1">
    <name>Multifunction Rake ZYX</name>
    <price>69.90</price>
    <shipDate>2001-01-24</shipdate>
  </item>
</purchaseOrder>
```

Verwenden des Standarddokumentmodells

Wenn in der Anweisung CREATE INDEX kein Dokumentmodell angegeben ist, verwendet Net Search Extender das Standarddokumentmodell.

Informationen zu diesem Vorgang

Ein Merkmal des Standarddokumentmodells ist, dass Abschnittsnamen in XPath-Schreibweise vorliegen, wodurch der absolute Pfad zu den einzelnen Elementen und Attributen angegeben wird. Beachten Sie, dass Abschnittsnamen in der Suchabfrage keine XPath-Ausdrücke sind, die bei der Abfrageausführung ausgewertet werden. Stattdessen handelt es sich um Namen, die auf bestimmte Teile (Elemente und Attribute) in strukturierten Dokumenten verweisen.

Wenn Sie keine Modelldatei verwenden, definieren Sie einen Textindex für XML-Dokumente wie folgt:

```
db2text CREATE INDEX i1 FOR TEXT ON t1(c2) CONNECT TO mydbname
```

Da die Spalte c2 den Datentyp XML aufweist, können Sie die Spezifikation FORMAT auslassen. Die Spezifikation FORMAT wird in diesem Fall automatisch auf XML gesetzt.

Wenn kein Dokumentmodell angegeben ist, wird jedem XML-Element in Abhängigkeit vom absoluten XPath im Dokument ein Name zugeordnet. Beispiel: Auf das Element price kann mit dem Abschnittsnamen /purchaseOrder/item/price in der Suchabfrage zugegriffen werden. Auf das Attribut countryCode kann mit dem Abschnittsnamen /purchaseOrder/shipAddress/@countryCode zugegriffen werden.

Wenn der Index mit dem Befehl **db2text update** aktualisiert wurde, lautet ein möglicher SQL-Ausdruck mit der Suchfunktion SECTION und der skalaren Suchfunktion wie folgt:

```
SELECT c2 FROM t1
WHERE CONTAINS(c2, SECTIONS("/purchaseOrder/item/name") "Rake") = 1
```

Die Abfrage gibt das oben angezeigte XML-Beispieldokument zurück.

Verwenden eines angepassten Dokumentmodells

Wenn Sie angepasste Abschnittsnamen definieren möchten, müssen Sie eine Modelldatei angeben, mit der benutzerdefinierte Namen bestimmten Teilen eines Dokuments zugeordnet werden. Ein Vorteil der Verwendung eines Dokumentmodells ist, dass Sie angeben können, welche Teile eines XML-Dokuments Sie indexieren möchten, und XPath-Ausdrücke zum Angeben dieser Teile verwenden können.

Informationen zu diesem Vorgang

Eine Modelldatei für das oben beschriebene XML-Dokument kann wie folgt aussehen:

```
<?xml version="1.0"?>
<XMLModel>
  <XMLFieldDefinition
    name="itemName"
    locator="/purchaseOrder/item/name" />
  <XMLFieldDefinition
    name="customerName"
    locator="//shipAddress/name" />
  <XMLAttributeDefinition
    name="partNumber"
    type="NUMBER"
    locator="/purchaseOrder//item/partNo" />
  <XMLFieldDefinition
    name="none"
    locator="/purchaseOrder/orderDate"
    exclude="yes" />
</XMLModel>
```

Beachten Sie, dass das Dokumentmodell den Namen itemName zum Element /purchaseOrder/item/name zuordnet, auf das in der vorherigen Suchabfrage verwiesen wird.

Die Indexdefinition mit der Modelldatei lautet wie folgt:

```
CREATE INDEX i1 FOR TEXT ON t1(c2) DOCUMENTMODEL XMLModel IN
/mydir/myfilename/xmlmodel.xml CONNECT TO mydbname
```

Der Name des Dokumentmodells (unter Verwendung des Parameters DOCUMENTMODEL) gibt das Stammelement in der Modelldatei an. Für XML-Dokumentmodelle lautet er XMLModel. Der Pfad /mydir/ ... verweist auf die Datei, die das Modell definiert.

Die Syntax des Dokumentmodells unterstützt eine Untergruppe der XPath-Syntax W3C, mit der eine bequeme Identifikation von Elementen möglich ist.

Nach der Erstellung des Textindex mit der oben stehenden Modelldatei und nach der Aktualisierung des Index mit dem Befehl **db2text update** kann nach dem Element /purchaseOrder/item/name wie folgt gesucht werden:

```
SELECT c2 FROM t1
WHERE CONTAINS(c2, SECTIONS("itemName") "Rake") = 1
```

Beachten Sie den Unterschied zur Suchabfrage, bei der kein Dokumentmodell angegeben wurde. Beide Abfragen geben dasselbe oben angegebene XML-Beispieldokument zurück.

Das XML-Dokumentmodell definiert auch ein Attribut partNumber für das XML-Attribut partNo des Elements item. Der Datentyp von Net Search Extender-Attributdefinitionen muss immer NUMBER lauten.

Die Attributdefinition in der obigen Beispielmodelldatei ermöglicht die Suche in Wertebereichen wie z. B.:

```
SELECT c2 FROM t1 WHERE CONTAINS
(c2, ATTRIBUTE "partNumber" BETWEEN 300 AND 500) = 1
```

XQuery-Unterstützung

Beim Suchen nach XML-Dokumenten in der Datenbank können die Suchergebnisse auch mit XQuery verarbeitet werden. Wenn Sie die Hybriddatenbanksteuerkomponente des DB2-Datenbankservers verwenden, kann eine SQL-Textsuchabfrage mit der XQuery-Verarbeitung kombiniert werden.

Informationen zu diesem Vorgang

Dazu wird die Eingabefunktion db2-fn:sqlquery() im XQuery-Kontext verwendet. Wenn Sie die XQuery-Eingabefunktion verwenden möchten, müssen Sie von SQL zu XQuery wechseln und dabei den Befehl **set language XQuery** verwenden; oder es muss das Schlüsselwort XQuery vor der Abfrage stehen. Hierbei handelt es sich um einen wichtigen Anzeiger für den Parser, dass die Ausführung mit einem XQuery-Ausdruck stattfindet; Sie müssen die Groß-/Kleinschreibungsregeln sowie die Syntaxregeln beachten, die für die XQuery-Sprache gelten.

Die Funktion db2-fn:sqlquery() übernimmt ein Zeichenfolgeliteral, das einen Fullselect darstellt. Die Funktion db2-fn:sqlquery() gibt eine XML-Folge zurück, die die Verkettung der XML-Spaltenwerte darstellt, die durch den Fullselect ausgewählt werden.

Der folgende Ausdruck kann zum Kombinieren von Testsuche und XQuery-Verarbeitung für nativ gespeicherte XML-Dokumente verwendet werden:

```
XQUERY db2-fn:sqlquery('SELECT c2 FROM t1
                        WHERE CONTAINS(c2,
                        ''SECTIONS ("/purchaseOrder/item/name") "Rake" '')
                        = 1 '//shipAddress/name
```

Diese Abfrage gibt alle name-Elemente unter dem Element shipAddress in XML-Dokumenten zurück, die eine Bestellposition mit dem Namen "Rake" enthalten. Sie müssen die XML-Spalte (in unserem Fall c2) in der SELECT-Anweisung explizit auswählen.

Das obige Beispiel kann durch ein FLWOR-Konstrukt wie folgt erweitert und in Ihre Anwendung integriert werden:

```
XQUERY FOR $item in db2-fn:sqlquery('SELECT c2 FROM t1
    WHERE CONTAINS(c2, ' SECTIONS ("/purchaseOrder/item/name") "Rake" ''
    = 1 ')
    WHERE $item[@partNo > "800"]
RETURN $item/price
```

Beachten Sie, dass der Fullselect der Eingabefunktion db2-fn:sqlquery() immer das vollständige XML-Dokument zurückgibt, für das ein Treffer zutrifft.

Ziehen Sie das folgende XML-Dokument zu Rate, das nativ in der Datenbank gespeichert ist:

```
<?xml version="1.0"?>
<dept bldg="101">
  <employee id="901">
    <name>Sabine</name>
    <resume>DB2 programmer</resume>
  </employee>
  <employee id="902">
    <name>Holger</name>
    <resume>XML expert</resume>
  </employee>
</dept>
```

Die Suche nach einem Mitarbeiter (employee) in Ihrer Abteilung, wobei der Begriff "XML" im Lebenslauf (resume) enthalten ist, kann wie folgt aussehen:

```
SELECT c2 FROM t1 WHERE CONTAINS(c2, SECTIONS("/dept/employee/resume") "XML")=1
```

Diese SELECT-Anweisung gibt das vollständige XML-Dokument zurück. Sie können die Suchabfrage in XQuery wie folgt einbetten:

```
XQUERY db2-fn:sqlquery('SELECT c2 FROM t1
    WHERE CONTAINS(c2,
    ''SECTIONS ("/dept/employee/resume") "XML" '' =1') //employee/name
```

Daraufhin werden die folgenden beiden Ergebnisse zurückgegeben:

```
<name>Sabine</name>
<name>Holger</name>
```

Beachten Sie, dass zwar der Begriff "XML" nicht im Lebenslauf der Mitarbeiterin Sabine auftritt, sie aber dennoch in der Ergebnisfolge dieser XQuery-Abfrage aufgeführt wird. Dazu kommt es, weil der Fullselect das gesamte Dokument zurückgibt, d. h., es wird das gesamte XML-Dokument zurückgegeben, in dem mindestens ein Mitarbeiter mit dem Begriff "XML" im Lebenslauf enthalten ist.

Wenn die Abfrage nur das Ergebnis <name>Holger</name> zurückgeben soll, setzen Sie die folgende XQuery-Anweisung ab:

```
XQUERY for $d in db2-fn:sqlquery('SELECT c2 FROM t1
    WHERE CONTAINS(c2,
    ''SECTIONS ("/dept/employee/resume") "XML" '' =1')
    return $d/dept/employee/name[contains(parent::employee/resume,"XML")];
```

Net Search Extender filtert alle XML-Dokumente aus, die den Begriff XML im Abschnitt /dept/employee/resume aufweisen; dabei wird ein strukturempfindlicher Volltextindex für die XML-Spalte verwendet. Auf der Basis der zurückgegebenen Untergruppe von XML-Dokumenten gibt die Rückkehranweisung `return §d/dept/employee/name[contains(parent::employee/resume,"XML")]` nur die <name>-Elemente zurück, die über XML in ihrem gleichgeordneten Element mit dem Namen <resume> verfügen; dazu wird das XML-Dokument mit der XPath-Achse durchsucht.

Kapitel 53. Unterstützung für strukturierte Dokumente

Beschreibung strukturierter Dokumente durch ein Dokumentmodell

Dokumente in HTML- oder XML-Format sind Beispiele für strukturierte Dokumente. Sie enthalten Tags, die wiederum Textfelder und Dokumentattribute enthalten. Textfelder können Informationen wie Titel, Autor oder eine Beschreibung des Dokuments enthalten.

Im Folgenden sehen Sie einen Auszug aus einem strukturierten reinen Textdokument. Es enthält Textelemente, die durch HTML-ähnliche Tags begrenzt sind.

```
[head]Handhabung strukturierter Dokumente  
[/head]
```

```
[abstract]Dieses Dokument beschreibt das Konzept strukturierter Dokumente  
und die Verwendung von Dokumentmodellen...  
[/abstract]:  
:
```

Wenn Net Search Extender strukturierte Dokumente indiziert, muss die Struktur erkannt werden, sodass das Textfeld und die Attribute indiziert und gemeinsam unter einem eindeutigen Namen gespeichert werden können. Net Search Extender hat dadurch die Möglichkeit, mit der Klausel SECTION bzw. ATTRIBUTE in einem bestimmten Textfeld eine selektive Suche durchzuführen oder nach Dokumenten mit einem bestimmten Attribut zu suchen.

Damit Net Search Extender die Struktur eines bestimmten Dokumentformats verstehen kann, müssen Sie eine Definition der Struktur in einem *Dokumentmodell* an Net Search Extender übergeben. Alternativ dazu können Sie die in Net Search Extender enthaltenen Standarddokumentmodelle verwenden.

Wenn Sie den Befehl **CREATE INDEX** zum Indexieren der Dokumente aufrufen, geben Sie den Namen des Dokumentmodells als Argument an. Beispiel: `CREATE INDEX i1 FOR TEXT ON t1(c2) DOCUMENT MODEL GPPModel IN mymodel.x1d CONNECT TO db`

Der Parameter **GPPModel** bezieht sich auf den verwendeten Typ des Dokumentmodells.

Bevor Sie Dokumente mit einem Dokumentmodell indizieren können, müssen Sie zunächst ein Dokumentmodell definieren und dieses dann dem Index bereitstellen.

Anmerkung: Wenn XML-Dokumente Indizes verwenden, die nicht korrekt formatiert sind, wird der Indexierungsprozess an der Stelle gestoppt, an der der Fehler im Dokument festgestellt wird. So wird nur ein Teil des Dokuments indiziert. Wenn Sie das Dokument nicht korrigieren, können nur die Teile des Dokuments durchsucht werden, die indiziert wurden. Dies tritt nur dann auf, wenn es sich beim verwendeten Tabellenspalten Typ nicht um XML handelt.

Beispiel für ein Dokumentmodell

Ein Dokumentmodell besteht aus Textfeld- und Attributdefinitionen.

Sie müssen für jedes Dokumentformat, das Sie indexieren möchten, ein Dokumentmodell definieren. Im Folgenden ist ein einfaches Dokumentmodell für strukturierte reine Textdokumente aufgeführt. Beachten Sie, dass im unten stehenden Beispiel GPP die Abkürzung für 'General Purpose Parser' ist.

```
<?xml version="1.0"?>
<GPPModel>                                - Hier beginnt das GPP-Dokumentmodell

  <GPPFieldDefinition                       - Hier beginnt eine Felddefinition
    name="Head"                             - Der Name, den Sie diesem Feld zuordnen
    start="[head]"                          - Die Begrenzungszeichenfolge am Anfang des Felds
    end="[/head]"                            - Die Begrenzungszeichenfolge am Ende des Felds
    exclude="YES" />

  <GPPFieldDefinition                       - Hier beginnt die nächste Felddefinition
    name="Abstract"
    start="[abstract]"
    end="[/abstract]"
    exclude="NO" />
:
:
</GPPModel>
```

Dokumentmodelle werden in der XML-Sprache mit Tags angegeben, die in Kapitel 54, „Dokumentmodellreferenz“, auf Seite 199 definiert werden. Ein Dokumentmodell besteht aus Textfeld- und Attributdefinitionen. Das oben stehende Beispiel demonstriert nur Textfelddefinitionen, die in GPPFieldDefinition-Elementen definiert werden. Ähnlich können Sie auch GPPAttributeDefinition-Elemente zum Definieren von Dokumentattributen verwenden.

Die erste Zeile des Beispiels, <?xml version="1.0"?>, gibt an, dass im Dokumentmodell XML-Tags verwendet wurden. Jede Textfelddefinition gibt Begrenzungszeichenfolgen an, um Start (start) und Ende (end) der Felddefinition im Quelldokument zu kennzeichnen. Wenn also in einem Dokument Textelemente zwischen der Begrenzungszeichenfolge [head] und der Begrenzungszeichenfolge [/head] stehen, wird dieser Text als Inhalt des Textfelds mit dem Namen head erkannt.

Jeder Felddefinition ordnen Sie einen Feldnamen zu. Durch diesen Feldnamen kann eine Abfrage die Suche auf den Inhalt eines Textfelds mit einer SECTION-Klausel in der Funktion CONTAINS beschränken. Der Name des Felds kann entweder festgelegt sein oder durch eine Regel vom Inhalt der strukturellen Einheit abgeleitet werden. Ein solcher Name kann z. B. der Tagname einer XML-Entität oder der Name eines XML-Attributs sein.

Dokumentmodelle

Ein Dokumentmodell steuert in erster Linie, welche Teile der Struktur eines Dokuments indexiert werden müssen und wie die Indexierung erfolgt.

Es dient folgenden Zwecken:

- Identifizieren von Textfeldern, die im Quelldokument zu unterscheiden sind
- Bestimmen des Typs eines solches Textfeldes
- Zuordnen eines Feldnamens zu einem Textfeld

Wenn das Dokumentmodell Text als zu einem Textfeld gehörig definiert, wird der Text als Teil des Textinhalts des Dokuments betrachtet, sodass Begriffe extrahiert und im Index gespeichert werden.

Die Elemente eines Dokumentmodells variieren je nachdem, welcher Parser für das betreffende Dokumentformat verwendet wird:

- Für das HTML-Format verwendet ein Dokumentmodell die HTML-Tagnamen, um zu definieren, welche HTML-Tags zu indexieren sind und wie die Metataginformationen zu behandeln sind.
- Für das XML-Format gibt es keinen vordefinierten Satz von Tags, sodass ein Dokumentmodell zunächst die relevanten Tags definieren muss. XML-Elemente des gleichen Namens können auch durch die Umgebung der anderen Elemente, in die sie eingebettet sind, unterschieden werden.
- Für das GPP-Format (GPP - General Purpose Parser) interagiert das Dokumentmodell noch intensiver mit dem Parser, weil es die Grenzen der Textfelder festlegen muss. In diesem Format muss die Felddefinition Zeichenfolgen angeben, die eine Erkennung der Grenzen von Feldern ermöglichen.
- Für Outside In-Formate verwendet ein Dokumentmodell mit den HTML-Tagnamen vergleichbare Tags, um zu definieren, welche Tags zu indexieren und wie die Metataginformationen zu bearbeiten sind. Beachten Sie hierbei, dass die Outside In Transformation Technology auch als INSO-Format bezeichnet wird.

Textfelder

Mit einem Dokumentmodell können Sie Abschnitte von Dokumenten angeben, die entweder zu einem bestimmten Textfeld gehören oder ein Dokumentattribut darstellen sollen, oder beides.

Der Text eines Dokuments wird vollständig indexiert, ungeachtet dessen, ob er Teil eines Textfelds ist oder nicht. Aussagekräftige Begriffe werden extrahiert und im Index gespeichert. Dies bedeutet, dass bei uneingeschränkten Textsuchvorgängen auch dieser Text durchsucht wird.

Allerdings können Sie durch die Definition von Textfeldern in einem bestimmten Feld selektiv nach Textelementen suchen. Sie können z. B. im Textfeld Abstract nach Dokumenten suchen, die das Wort structure enthalten. Beispiel: `SELECT doc from my_docs WHERE CONTAINS (doc, SECTIONS(abstract) "structure" = 1.`

Ein Textfeld kann in einem Dokument mehr als einmal auftreten. Sie können z. B. ein Textfeld definieren, das alle Abbildungsunterschriften enthält. Ein Textfeld kann sogar ein anderes Textfeld überlappen.

Wenn Sie den Inhalt bestimmter Textfelder nicht indexieren möchten, können Sie eine Felddefinition angeben, die `exclude="YES"` enthält. Unter „Begrenzungen für Textfelder und Dokumentattribute“ auf Seite 202 finden Sie eine Liste mit Begrenzungen für Textfelder und Dokumentattribute.

Dokumentattribute

Dokumentattribute enthalten kurze, formatierte Informationen des Typs number.

Im Gegensatz zu Textfeldern können Sie zum Durchsuchen von Dokumenten mit solchen Attributen Wertebereiche verwenden.

Attribute werden nicht mit indexierten Textelementen, sondern in einem separaten Elementindex gespeichert. Wenn Sie also auf der Basis des Inhalts eines Attributs nach Dokumenten suchen möchten, müssen Sie eine Attributsuche explizit für das Attribut durchführen. Beispiel: `SELECT doc FROM my_docs WHERE CONTAINS (doc,ATTRIBUTE "year" BETWEEN 2001 AND 2005) = 1.`

Zahlenattribute

Net Search Extender stellt einen Parser zur Verfügung, der Gleitkommazahlen erkennt.

Im Folgenden sehen Sie Beispiele korrekter und inkorrektter Formate für Attributwerte.

Tabelle 1. Unterstützte Formate für Attributwerte

Richtiges Format	Falsches Format
1000 1 000 1.000 - Hierbei ist der Punkt ein Dezimalzeichen	1,000
100 000 100 000.00123	1 000 000 - Zwei Leerzeichen zwischen 1 und 0

Beachten Sie, dass keine Leerzeichen im Dezimalbereich einer Zahl erlaubt sind. So wird z. B. die Zahl 1 000.000 100 wie zwei separate Zahlen behandelt: 1000.000 und 100.

Sprachspezifische Trennzeichen und länderspezifische Währungsformate werden nicht unterstützt.

Standarddokumentmodelle

Bei der Verwendung eines der Standarddokumentmodelle ist Folgendes zu beachten: Alle Felder werden indiziert, es werden keine speziellen Informationen extrahiert und es werden keine numerischen Attribute indiziert.

Für HTML- und XML-Dokumente sowie für Outside In-Filterdokumente stellt Net Search Extender Standarddokumentmodelle bereit, die verwendet werden, wenn Sie bei der Indexerstellung kein Dokumentmodell angeben. Für strukturierte einfache Textdokumente müssen Sie ein Dokumentmodell bereitstellen und angeben.

Wenn Sie eines der Standarddokumentmodelle verwenden, gelten folgende Regeln:

- Alle Felder werden indiziert, wobei keine Sonderinformationen, wie zum Beispiel Metainformationen, extrahiert werden.
 - Bei HTML- und INSO-Formaten wird jedem Feld der Name des entsprechenden Tags zugeordnet.
 - Bei XML werden alle XML-Knoten eines XML-Dokuments überlappenden Feldern zugeordnet, die durch die vollständig qualifizierten Elementpfade der entsprechenden Knoten angegeben werden. Beispiel: der Pfad `/play/role/name`.
- Es werden keine numerischen Attribute indiziert (da im Standarddokumentmodell keine numerischen Attribute definiert sind).

Tabelle 2. Funktionsweisen der Standarddokumentmodelle für die unterstützten Dokumentformate

Dokumenttyp	Funktionen des Standarddokumentmodells
HTML	Akzeptiert die folgenden Tags als Textfelder: <code><a></code> <code><address></code> <code><au></code> <code><author></code> <code><h1></code> <code><h2></code> <code><h3></code> <code><h4></code> <code><h5></code> <code><h6></code> <code><title></code> . Feldname ist der Tagname, zum Beispiel 'address'.

Tabelle 2. Funktionsweisen der Standarddokumentmodelle für die unterstützten Dokumentformate (Forts.)

Dokumenttyp	Funktionen des Standarddokumentmodells
XML	Akzeptiert alle Tags als Textfelder. Der Feldname ist der vollständig qualifizierte Name des Elementpfads, z. B. '/play/title'.
Strukturierte Textdatei (GPP)	Kein Standarddokumentmodell vorhanden.
Outside In (INSO)	Akzeptiert als Textfelder die Dokumenteigenschaften unter „Definition eines Dokumentmodells für mit Outside In gefilterte Dokumente“ auf Seite 196, die von den Outside In-Filtern zurückgegeben werden. Der Feldname ist der von Outside In verwendete Name der Dokumenteigenschaft, z. B. 'SCCCA_TITLE'.

Für jeden Dokumenttyp wird ein Standarddokumentmodell definiert. Da jedes Modell anders ist, wird in den folgenden Abschnitten für jedes Modell ein Beispiel mit Erläuterung bereitgestellt.

Anmerkung:

Obwohl die Standarddokumentmodelle Dokumente korrekt verarbeiten, sollten Sie zur besseren Indexierung und Suche eigene Dokumentmodelle definieren.

Mit dem Standarddokumentmodell wird der Text eines Dokuments vollständig indexiert, ungeachtet dessen, ob er Teil eines Textfelds ist oder nicht. Dies bedeutet, dass bei uneingeschränkten Textsuchvorgängen auch dieser Text durchsucht wird.

Definition eines Dokumentmodells für strukturierte reine Textdokumente

Dieser Abschnitt enthält die Parameter der Dokumentmodellelemente sowie Informationen zu Einschränkungen.

Die folgenden Parameter der Dokumentmodellelemente werden verwendet:

name Sie ordnen dem Textfeld bzw. Dokumentattribut für jede Definition einen Namen zu. Mithilfe der Namen können Sie eine Suchabfrage auf den Inhalt eines bestimmten Textfelds oder eines Dokumentattributs begrenzen. In den oben gezeigten Beispielen könnten Sie zum Beispiel nach Dokumenten suchen, die das Wort `structure` im Textfeld mit dem Namen `Abstract` enthalten.

start Eine Begrenzungszeichenfolge in Codepage UTF-8, die den Anfang des Textfelds oder Dokumentattributs markiert. Für die Angabe von Begrenzungszeichenfolgen gibt es keine Regeln. Es kann jede beliebige UTF-8-Zeichenfolge verwendet werden. Einige Beispiele: `start="introduction:"`, `start="note!"`, `start="$$. ."`.

Nicht druckbare Zeichen und die XML-Sonderzeichen '<' und '&' müssen mithilfe der XML-Standardzeicheneingabe ('<' für '<' und '&' für '&') angegeben werden.

end Optional. Eine Begrenzungszeichenfolge in der Codepage UTF-8, die das Ende des Textfelds oder Dokumentattributs markiert. Wenn Sie keinen Endtag angeben, wird der nächste gefundene Starttag als Ende des Feldes

angenommen. Wenn kein nachfolgender Starttag gefunden wird, erstreckt sich das Feld bis zum Ende des Dokuments, und es werden keine weiteren Felder erkannt.

type Der Typ eines Dokumentattributs muss immer 'NUMBER' sein. Dieser Parameter hat für Felddefinitionen keine Gültigkeit.

exclude YES oder NO. Ein Parameter, der festlegt, ob der Text in einer Felddefinition ausgeschlossen und somit nicht indexiert werden soll. Dieser Parameter hat für Attributdefinitionen keine Gültigkeit.

In dem gezeigten Beispiel würde die Felddefinition 'head' ausgeschlossen, während die Definition 'abstract' mit eingeschlossen würde.

Einschränkung:

- Es dürfen keine zwei Felddefinitionen bzw. Attributdefinitionen mit dem gleichen Starttag vorhanden sein. Allerdings dürfen eine Felddefinition und eine Attributdefinition die gleichen Start- und Endtags haben.
- Ein Starttag darf kein Präfix eines anderen Starttags sein. Zum Beispiel ist es nicht möglich, einen Starttag 'author' und einen Starttag 'authority' zu definieren.
- Starttags und Endtags dürfen keine leeren Zeichenfolgen sein.

Beim Indexieren eines GPP-Dokuments ausgeführte Operationen

Der GPP-Parser durchsucht das Dokument nach einer der Startbegrenzungszeichenfolgen. Wenn er eine Startbegrenzung findet, analysiert er das nachfolgende Feld, bis er die entsprechende Endbegrenzungszeichenfolge findet.

Der Inhalt des Felds wird anschließend gemäß dem Definitionsterm, d. h. als Textfeld oder als Dokumentattribut, indexiert. Wenn das Textfeld und das Dokumentattribut die gleichen Begrenzungszeichenfolgen für Anfang und Ende haben, wird der Inhalt des Feldes sowohl als Textfeld als auch als Dokumentattribut indexiert.

Eine Verschachtelung von Feldern ist nicht zulässig. Wenn eine neue Startbegrenzungszeichenfolge in einem Feld angetroffen wird, bevor die Endzeichenfolge erreicht wurde, wird die neue Startbegrenzungszeichenfolge als normaler Text interpretiert.

Wenn keine entsprechende Endzeichenfolge gefunden wird, nimmt der Parser an, dass das Feld bis zum Ende des Dokuments reicht, und es wird ein entsprechender Ursachencode gemeldet.

Wenn im Dokumentmodell keine Endbegrenzungszeichenfolge definiert wird, markiert die neue Startbegrenzungszeichenfolge das Ende des vorigen Felds.

Definition eines Dokumentmodells für HTML-Dokumente

Der HTML-Parser wandelt den Text in die Codepage UTF-8 um. Er führt eine HTML-Tagerkennung durch und klassifiziert die Tags in Tagklassen.

Der HTML-Parser wandelt den Text in die Codepage UTF-8 um. Er führt eine HTML-Tagerkennung durch und klassifiziert die Tags in Tagklassen:

- Mit Tags ausgezeichnete Informationen, die zu ignorieren sind, wie zum Beispiel Schriftartinformationen

- Tags, die Positionsinformationen enthalten, wie zum Beispiel <p> für neuen Absatz.
- Tags, die Strukturinformationen enthalten, wie <Title>

Er erkennt alle Verweise auf Zeichenentitäten, die in HTML 4 definiert sind, wie 'ä' (ä), und löst sie in die entsprechenden Codepunkte in UTF-8 auf.

Er erkennt Metatags und analysiert den Metatagtext.

Das folgende Beispiel zeigt ein HTML-Dokument:

```
<HTML>
<HEAD>
<META NAME="year" CONTENT="2002">
<TITLE> The Firm </TITLE>
</HEAD>
<BODY>
<H1>Synopsis</H1>;
<H1>Prologue</H1>;:
:
</BODY>
```

Das folgende Beispiel zeigt ein HTML-Dokumentmodell:

```
<?xml version="1.0"?>
<HTMLModel>

  <HTMLFieldDefinition
    name="subtitle"
    tag="title"
    exclude="NO" />

  <HTMLFieldDefinition      - Dies ist der Anfang eines Textfelds
    name="header1"
    tag="h1"
    exclude="YES" />      - Dies ist das Ende des Textfelds

  <HTMLAttributeDefinition - Dies ist der Anfang des Dokumentattributs
    name="year"
    tag="meta"
    meta-qualifier="year"
    type="NUMBER" />      - Dies ist das Ende des Dokumentattributs

</HTMLModel>
```

Die erste Zeile, <?xml version="1.0"?>, gibt an, dass das Modell unter Verwendung von XML-Tags geschrieben ist. Beachten Sie, dass dieses Modell nicht für Dokumente im XML-Format geschrieben ist.

Jedes Feld wird innerhalb eines Tags HTMLFieldDefinition oder HTMLAttributeDefinition definiert, das Elementparameter enthält.

Alle Textfelddefinitionen müssen im Tag <HTMLModel> enthalten sein. Der Tagname wird bei der Indexerstellung als Parameter angegeben: CREATE INDEX iA FOR TEXT ON T1(C2) DOCUMENTMODEL HTMLModel IN myModel.xml CONNECT TO db.

Die folgenden Parameter der Dokumentmodellelemente werden verwendet:

name Sie ordnen dem Textfeld bzw. Dokumentattribut für jede Definition einen Namen zu. Mithilfe der Namen können Sie eine Suchabfrage auf den Inhalt eines bestimmten Textfelds oder eines Dokumentattributs begrenzen.

In den oben gezeigten Beispielen könnten Sie zum Beispiel nach Dokumenten suchen, die das Wort `firm` im Textfeld mit dem Namen `subtitle` enthalten.

tag Gibt ein Element an, dessen Start- und (implizierte) Endtags das Textfeld bzw. das Dokumentattribut markieren. Der Text, der sich innerhalb eines Elements dieses Namens befindet, stellt den Inhalt des definierten Felds dar.

Die Groß-/Kleinschreibung des Tags wird ignoriert.

In den oben gezeigten Beispielen wird der Text, der einem Tag `H1` folgt, als Teil des Felds `'header1'` indexiert. Dem Beispieldokument entsprechend würden die Wörter `'synopsis'` und `'prologue'` indexiert.

meta-qualifier

Dieser Tag muss zusammen mit dem Element **tag** verwendet werden. Durch die Angabe `tag="meta"` wird der Wert des Inhalts (CONTENT) extrahiert, dessen Wert durch meta-qualifier angegeben ist.

Im Beispiel für ein HTML-Dokument enthält der Metatag die folgenden Elemente:

```
<META NAME="year" CONTENT="2002">
```

Das Beispiel des Dokumentmodells definiert das Element `meta-qualifier="year"`. Der Inhalt `'2002'` wird deshalb als Wert des Attributs `'year'` indexiert.

type Der Typ eines Dokumentattributs muss `'NUMBER'` sein. Dieser Parameter hat für Felddefinitionen keine Gültigkeit.

exclude

YES oder NO. Ein Parameter, der festlegt, ob der Text in einer Felddefinition ausgeschlossen und somit nicht indexiert werden soll. Dieser Parameter hat für Attributdefinitionen keine Gültigkeit.

In dem gezeigten Beispiel würde die Felddefinition `'header1'` ausgeschlossen, während die Definition `'subtitle'` mit eingeschlossen würde.

Der gesamte übrige Text eines Dokuments wird indexiert, jedoch nicht als Teil eines Felds.

Elementparameter

Die folgenden Parameter der Dokumentmodellelemente werden verwendet:

name Sie ordnen dem Textfeld bzw. Dokumentattribut für jede Definition einen Namen zu. Mithilfe der Namen können Sie eine Suchabfrage auf den Inhalt eines bestimmten Textfelds oder eines Dokumentattributs begrenzen. In den oben gezeigten Beispielen könnten Sie zum Beispiel nach Dokumenten suchen, die das Wort `firm` im Textfeld mit dem Namen `subtitle` enthalten.

tag Gibt ein Element an, dessen Start- und (implizierte) Endtags das Textfeld bzw. das Dokumentattribut markieren. Der Text, der sich innerhalb eines Elements dieses Namens befindet, stellt den Inhalt des definierten Felds dar.

Die Groß-/Kleinschreibung des Tags wird ignoriert.

In den oben gezeigten Beispielen wird der Text, der einem Tag H1 folgt, als Teil des Felds 'header1' indexiert. Dem Beispieldokument entsprechend würden die Wörter 'synopsis' und 'prologue' indexiert.

meta-qualifier

Dieser Tag muss zusammen mit dem Element **tag** verwendet werden. Durch die Angabe `tag="meta"` wird der Wert des Inhalts (CONTENT) extrahiert, dessen Wert durch meta-qualifier angegeben ist.

Im Beispiel für ein HTML-Dokument enthält der Metatag die folgenden Elemente:

```
<META NAME="year" CONTENT="2002">
```

Das Beispiel des Dokumentmodells definiert das Element `meta-qualifier="year"`. Der Inhalt '2002' wird deshalb als Wert des Attributs 'year' indexiert.

type Der Typ eines Dokumentattributs muss 'NUMBER' sein. Dieser Parameter hat für Felddefinitionen keine Gültigkeit.

exclude

YES oder NO. Ein Parameter, der festlegt, ob der Text in einer Felddefinition ausgeschlossen und somit nicht indexiert werden soll. Dieser Parameter hat für Attributdefinitionen keine Gültigkeit.

In dem gezeigten Beispiel würde die Felddefinition 'header1' ausgeschlossen, während die Definition 'subtitle' mit eingeschlossen würde.

Der gesamte übrige Text eines Dokuments wird indexiert, jedoch nicht als Teil eines Felds.

Definition eines Dokumentmodells für XML-Dokumente

Mit einem Dokumentmodell für XML-Dokumente können Sie definieren, wie ein Element, das in einem XML-Dokument gefunden wird, einem Feld und/oder einem Dokumentattribut zugeordnet wird.

Das folgende Beispiel zeigt ein XML-Dokument:

```
<?xml version="1.0"?>
<purchaseOrder orderDate="2001-01-20"> [4]
  <shipAddress countryCode="US"> [1]
    <name>Alice Smith</name> [2]
    <street>123 Maple Street</street>
    <city>Mill Hill</city>
    <state>CA</state>
    <zip>90999</zip>
  </shipAddress>
  <item partNo="123" quantity="1"> [3]
    <name>S&B Lawnmower Type ABC-x</name>
    <price>239.90</price>
    <shipDate>2001-01-25</shipDate>
  </item>
  <item partNo="987" quantity="1"> [3]
    <name>Multifunction Rake ZYX</name>
    <price>69.90</price>
    <shipDate>2001-01-24</shipDate>
  </item>
</purchaseOrder>
```

Das folgende Beispiel zeigt ein XML-Dokumentmodell, das mit dem oben stehenden Beispieldokument übereinstimmt:

```

<?xml version="1.0"?>
<XMLModel>

<XMLFieldDefinition [1]
name="addresses"
locator="/purchaseOrder/shipAddress" />

<XMLFieldDefinition [2]
name="customerName"
locator="//shipAddress/name"
exclude="yes"/>

<XMLAttributeDefinition [3]
name="partNumber"
type="NUMBER"
locator="/purchaseOrder//item/@partNo" />

<XMLFieldDefinition [4]
name="none"
locator="/purchaseOrder/@orderDate" />

</XMLModel>

```

Die erste Zeile, `<?xml version="1.0"?>`, gibt an, dass das Modell in XML geschrieben ist. Jedes Feld wird innerhalb eines Tags `XMLFieldDefinition` oder `XMLAttributeDefinition` definiert, das Elementparameter enthält.

Beachten Sie, dass alle Textfelddefinitionen in dem Tag `<XMLModel>` enthalten sein müssen. Dieser Tagname wird bei der Indexerstellung als Parameter angegeben:
`CREATE INDEX i1 FOR TEXT ON T1(C2) DOCUMENTMODEL XMLModel in myModel.xml`
`CONNECT TO db.`

Die in dem Beispiel enthaltenen Felder und Attribute sind mit Nummern markiert, die den Definitionen in der Beispielmotelldatei entsprechen.

Die Verschachtelung von Feldern ist beispielsweise dann erlaubt, wenn die XPath-Speicherposition einer einzigen Spezifikation einen Knoten auswählt, der sich innerhalb eines XML-Elements befindet, das von einer anderen Attributdefinition ausgewählt wurde. In dem oben stehenden XML-Beispieldokument sind verschachtelte Felder enthalten. Das Feld `addresses` wählt in dem XML-Dokument einen Knoten aus, der den durch das Feld `customerName` ausgewählten Knoten dominiert. Es besteht daher eine logische Zuordnung des Inhalts dieses eingebetteten Knotens zu beiden Feldern. Obwohl Textfelder sich überlappen können, wird der Text in diesen Feldern nur einmal indexiert. Im vorliegenden Beispiel wird durch die Suche mit einer Feldbeschränkung der Name `Alice Smith` sowohl im Feld `addresses` als auch im Feld `customerName` gefunden. Allerdings kann aufgrund der übereinstimmenden Semantik des Querverweisausdrucks nicht ein und derselbe XML-Knoten mehreren Feldern zugeordnet werden.

Net Search Extender versucht nicht, die Codepage eines XML-Dokuments zu erkennen. Es wird die DB2-Codepage verwendet.

Der Inhalt von Feldern wird durch folgende Regeln bestimmt:

- Für ein Feld, dessen Querverweis einem Kommentar, einer Verarbeitungsanweisung oder einem XML-Attribut entspricht, ist der Feldinhalt der eigentliche Kommentartext, Verarbeitungsanweisungstext oder Attributwerttext.

- Für ein Feld, das mit einem XML-Element oder dem Stammknoten übereinstimmt, besteht der Feldinhalt aus beliebigem Text aus einem beliebigen eingebetteten Element. Dies gilt nicht für Elemente, die den Feldern mit der Spezifikation `exclude="YES"` entsprechen.

Das Dokument muss korrekt formatiertes XML enthalten; es ist jedoch nicht erforderlich, dass im XML-Dokument eine Dokumenttypdefinition (DTD - Document Type Definition) angegeben wird. Es wird keine DTD-Prüfung oder Auflösung externer Entitäten durchgeführt. Net Search Extender gleicht lediglich das XML-Dokument mit dem Dokumentmodell ab. Interne Entitäten werden den XML-Anforderungen entsprechend ersetzt.

Elementparameter

Die folgenden Parameter der Dokumentmodellelemente werden verwendet:

name Sie ordnen dem Textfeld bzw. Dokumentattribut für jede Definition einen Namen zu. Mithilfe dieser Namen können Sie eine Suchabfrage auf den Inhalt eines bestimmten Textfelds oder eines Dokumentattributs begrenzen.

Sie können eine der folgenden Variablen in einem Namen verwenden. Die Variable wird durch eine Zeichenfolge ersetzt, die aus dem übereinstimmenden Element im Quelldokument generiert wird.

Variable

Wert

\$(NAME)

Der tatsächliche qualifizierte Name (QName) des XML-Elements, das dem XPath-Wert entspricht.

\$(LOCALNAME)

Der tatsächliche lokale Name (ohne Präfix) des XML-Elements, das dem XPath-Wert entspricht.

\$(PATH)

Der tatsächliche absolute Pfad als Folge von Schrägstrichen und Tags des XML-Elements, das dem XPath-Wert entspricht (als Name im Standarddokumentmodell verwendet).

type Der Typ eines Dokumentattributs muss 'NUMBER' sein. Dieser Parameter hat für Felddefinitionen keine Gültigkeit.

locator

Ausdrücke in der XPath-Sprache, welche die Teile von Quelldokumenten auswählen, die als Suchfelder zu verwenden sind.

Beim Schreiben einer XML-Dokumentmodelldatei müssen die qualifizierten Namen (QNames) in einem Querverweis (locator) mit einigen Tags im XML-Dokument übereinstimmen, da sonst keine Felder erkannt werden und die Abfragen in den Feldern kein Ergebnis ausgeben.

Im Folgenden sind Querverweisbeispiele aufgeführt.

purchaseOrder | salesOrder

Alle purchaseOrder- und salesOrder-Elemente

shipAddress

Alle shipAddress-Elemente

* Alle Elemente (dies ist die Kurzform von `child::*`; weitere Informationen siehe Syntax)

name/item

Alle `item`-Elemente, die ein direkt übergeordnetes Element `name` haben

purchaseOrder//item

Alle `item`-Elemente, die ein übergeordnetes Objekt `purchaseOrder` haben

/ Der Stammknoten

comment()

Alle Kommentarknoten

processing-instruction()

Alle Verarbeitungsinstruktionen

attribute::* (oder @*)

Alle Attributknoten

Ein Literal ist eine Zeichenfolge, die entweder in einfache oder in doppelte Anführungszeichen gesetzt ist. Eine exakte Definition von Terminaltoken finden Sie in den XML-Empfehlungen.

Die vom Net Search Extender-Dokumentmodell unterstützten XPath-Querverweise sind XSLT-Mustern (XML Stylesheet Language Transformation) ähnlich. Sie umfassen exakt die Teilmenge von XSLT-Mustern, die weder Vergleichselemente noch die Funktionen 'id' und 'key' noch die Knotentests 'text()' und 'node()' enthalten.

ignore YES oder NO. Dieser Parameter dient zur Angabe von Ausnahmen für das locator-Element.

Manchmal kann es sinnvoll sein, einen allgemeinen Querverweis wie * anzugeben, um die Knoten zu bezeichnen, die indexiert werden sollen. Sie können jedoch auch angeben, dass einige Knoten, die einem spezifischeren Querverweis entsprechen, nicht indexiert werden sollen.

Zur Formulierung dieser Angabe fügen Sie eine Felddefinition mit einem spezifischeren Querverweis für die Knoten ein, die beim Indexieren zu ignorieren sind. Anschließend erteilen Sie diesem Querverweis eine höhere Priorität als dem allgemeinen Querverweis (siehe folgender Abschnitt) und geben `ignore="yes"` an. Dadurch wird der Indexierungsfunktion mitgeteilt, dass sie keine Feldinformationen für die entsprechenden Knoten generieren darf.

Beachten Sie, dass in dem Fall, dass ein solcher zu ignorierender Knoten in einen Knoten eingebettet ist, für den ein Feld generiert wird, der Inhalt des zu ignorierenden Knotens indexiert wird, weil er ebenfalls zum Inhalt des Feld generierenden Knotens gehört.

priority

Eine Gleitkommazahl zwischen -1 und +1, die angibt, dass die Priorität einer Definition zu erteilen ist, die durch einen bestimmten Querverweis gefunden wird.

Wenn Sie keine Priorität angeben, werden die Standardprioritäten verwendet:

- Mehrere Alternativen, die durch `|` getrennt sind, werden als eine Gruppe von Definitionen, d. h. jeweils eine für jede Alternative, behandelt.
- Querverweise, die einem einzelnen Namen entsprechen. Das heißt, Querverweise der folgenden Formate haben jeweils die Standardpriorität 0:

- ChildOrAttributeAxisSpecifier QName
- ChildOrAttributeAxisSpecifier processing-instruction(Literal))
- Querverweise des Formats ChildOrAttributeAxisSpecifier QName:* haben die Standardpriorität -0.25.
- Andere Querverweise des Formats ChildOrAttributeAxisSpecifier NodeTest haben die Standardpriorität -0.5.
- Alle anderen Querverweise haben die Standardpriorität 0.5.

Beachten Sie, dass die Standardpriorität umso höher ist, je spezifischer ein Querverweis ist. Zum Beispiel ordnet der unspezifische Querverweis * der gefundenen Definition eine niedrige Priorität zu, während ein Name ein spezifischerer Querverweis ist, der eine höhere Priorität verleiht.

Beachten Sie außerdem, dass Sie für den Fall, dass ein Knoten mehr als einem Querverweis entspricht, die Auswahl der Definitionen festlegen können, indem Sie ihnen Prioritäten zuordnen. Die Definition mit der höchsten Priorität wird gewählt. Wenn zwei Definitionen dieselbe Priorität haben, wird die zuletzt definierte gewählt.

Diese Art der Konfliktauflösung stimmt mit der in XSLT (XML Stylesheet Language Transformation) verwendeten überein.

exclude

YES oder NO. Ein Parameter, der festlegt, ob der Text in einer Felddefinition ausgeschlossen und somit nicht indexiert werden soll. Dieser Parameter hat für Attributdefinitionen keine Gültigkeit.

In dem gezeigten Beispiel würde die Felddefinition 'customerName' ausgeschlossen, während die Definition 'addresses' mit eingeschlossen würde.

Anmerkung: Wenn mehrere Felddefinitionen mit demselben Querverweiswert, aber verschiedenen Namen vorhanden sind, ist nur die letzte Felddefinition wirksam. Alle vorherigen Felddefinitionen werden ignoriert.

Die folgende Modelldatei enthält eine Felddefinition.

```
<XMLFieldDefinition
name="from"
exclude="NO"
locator="/document/email/from" />
```

Nach dem Indexieren werden die Dokumente gefunden, wenn im Abschnitt "from" gesucht wird.

Im nächsten Schritt wird eine weitere Felddefinition am Ende der Modelldatei hinzugefügt.

```
<XMLFieldDefinition
name="from_last"
exclude="NO"
locator="/document/email/from" />
```

Nach der erneuten Erstellung und Aktualisierung des Index werden die Dokumente gefunden, wenn im Abschnitt "from_last" gesucht wird, nicht jedoch, wenn im Abschnitt "from" gesucht wird.

Definition eines Dokumentmodells für mit Outside In gefilterte Dokumente

Dokumentmodelle für das Outside In-Format weisen zahlreiche Ähnlichkeiten mit HTML-Dokumentmodellen auf. Sie ermöglichen Ihnen ebenfalls die Zuordnung von Strukturelementen, die durch eine bestimmte Gruppe von Tags definiert werden, zu Net Search Extender-Textfeldern und -Dokumentattributen.

Angenommen, Sie haben eine Reihe von Microsoft Word-Dokumenten und möchten die Dokumenteigenschaften 'title', 'subject' und 'keyword' als Felder sowie die Dokumenteigenschaften 'author' und 'category' als Dokumentattribute indexieren. Diese Zuordnung wird mit dem folgenden Beispiel für ein Outside In-Dokumentmodell erreicht:

```
<?xml version="1.0"?>
<INSOModel>

<INSOFieldDefinition
name="title"
tag="SCCCA_TITLE"/>

<INSOFieldDefinition
name="title"
tag="SCCCA_SUBJECT"/>

<INSOFieldDefinition
name="title"
tag="SCCCA_KEYWORDS"/>

<INSOAttributeDefinition
name="author"
tag="SCCCA_AUTHOR"
type="STRING"/>

<INSOAttributeDefinition
name="category"
tag="SCCCA_CATEGORY"
type="STRING"/>

</INSOModel>
```

Elementparameter

Die folgenden Parameter der Dokumentmodellelemente werden verwendet:

- name** Ein Name, den Sie dem Textfeld oder Dokumentattribut zuordnen. Einen Feldname ordnen Sie den verschiedenen Felddefinitionen und einen Attributnamen den verschiedenen Attributdefinitionen zu. Diese Namen sind das Mittel, mit dem eine Abfrage die Suche auf den Inhalt eines bestimmten Textfelds beschränken und nach Dokumenten mit einem bestimmten Attribut suchen kann.
- tag** Bezeichnet einen Tag, dessen Start- und (implizierte) Endelemente das Textfeld bzw. das Dokumentattribut markieren. Der Text, der sich innerhalb eines Elements dieses Namens befindet, stellt den Inhalt des definierten Felds oder Attributs dar. Die Groß-/Kleinschreibung des Tags wird ignoriert. Mögliche Werte werden im folgenden Absatz beschrieben.
- type** Als Typ des Dokumentattributs ist 'NUMBER', 'DATE' oder 'STRING' zulässig. Dieser Parameter hat für Felddefinitionen keine Gültigkeit.

exclude

YES oder NO. Ein Parameter, der festlegt, ob der Text in einer Felddefinition ausgeschlossen und somit nicht indexiert werden soll. Dieser Parameter hat für Attributdefinitionen keine Gültigkeit.

Outside In-Dokumentmodelle bestehen aus Feld- und Attributdefinitionen, die jeweils einen Namen und einen Tag definieren. Für Attributdefinitionen wird zusätzlich ein Typ benötigt, während Felddefinitionen optional eine 'exclude'-Markierung enthalten können. Ebenso wie bei HTML-Modellen definiert das Namensattribut einer solchen Definition den Namen des Net Search Extender-Felds oder -Attributs, dem der Dokumentteil zugeordnet werden soll. Dabei kann es sich um eine willkürliche UTF8-Zeichenfolge handeln. Weitere Informationen finden Sie in der Outside In Content Access Specification, Version 7.5.

Beim Indexieren eines Outside In-Dokuments ausgeführte Operationen

Standardmäßig wird der gesamte Text als nicht einem bestimmten Feld zugehörig indexiert.

Jedes Mal, wenn ein Starttag innerhalb des Textdatenstroms mit einem Definitionselement im momentan aktiven Dokumentmodell übereinstimmt, wird der Text zwischen dem Starttag und dem entsprechenden Endtag gemäß dem Definitionsterm behandelt, also z. B. als indexiertes Feld oder ausgeschlossenes Feld und/oder Attribut.

Wenn keine entsprechende Definition existiert, werden der Starttag und der zugehörige Endtag ignoriert.

Da Outside In-Filter das Format und die Codepage des Dokuments automatisch erkennen, hat die Angabe der ID für den codierten Zeichensatz (CCSID) keine Auswirkung. Wenn die Outside In-Filter das korrekte Format und die korrekte Codepage nicht ermitteln können, wird das Dokument als ASCII-Datei identifiziert.

Kapitel 54. Dokumentmodellreferenz

Net Search Extender ist eine nützliche Quelle für Informationen zu Dokumentmodellen.

Net Search Extender stellt die folgenden Referenzinformationen für Dokumentmodelle zur Verfügung:

- DTD für Dokumentmodelle
- Semantik von Querverweisausdrücken (XPath)
- Begrenzung für Textfelder und Dokumentattribute
- Attributwerte für Outside In-Tag

DTD für Dokumentmodelle

Der vorliegende Abschnitt enthält eine formale Beschreibung der Syntax von Dokumentmodellen in Form einer Dokumenttypdefinition (DTD):

Das folgende Beispiel zeigt eine formale Beschreibung der Syntax von Dokumentmodellen in Form einer Dokumenttypdefinition (DTD):

```
<!ELEMENT GPPModel (GPPFieldDefinition|GPPAttributeDefinition)+>
<!ELEMENT HTMLModel (HTMLFieldDefinition|HTMLAttributeDefinition)+>
<!ELEMENT XMLModel (XMLFieldDefinition|XMLAttributeDefinition)+>

<!ELEMENT GPPFieldDefinition EMPTY>
<!ATTLIST GPPFieldDefinition name CDATA #REQUIRED>
<!ATTLIST GPPFieldDefinition start CDATA #REQUIRED>
<!ATTLIST GPPFieldDefinition end CDATA #IMPLIED>
<!ATTLIST GPPFieldDefinition exclude (YES|NO) NO>

<!ELEMENT GPPAttributeDefinition EMPTY>
<!ATTLIST GPPAttributeDefinition name CDATA #REQUIRED>
<!ATTLIST GPPAttributeDefinition start CDATA #REQUIRED>
<!ATTLIST GPPAttributeDefinition end CDATA #REQUIRED>
<!ATTLIST GPPAttributeDefinition type NUMBER #REQUIRED>

<!ELEMENT HTMLFieldDefinition EMPTY>
<!ATTLIST HTMLFieldDefinition name CDATA #REQUIRED>
<!ATTLIST HTMLFieldDefinition tag CDATA #REQUIRED>
<!ATTLIST HTMLFieldDefinition meta-qualifier CDATA #IMPLIED>
<!ATTLIST HTMLFieldDefinition exclude (YES|NO) NO>

<!ELEMENT HTMLAttributeDefinition EMPTY>
<!ATTLIST HTMLAttributeDefinition name CDATA #REQUIRED>
<!ATTLIST HTMLAttributeDefinition tag CDATA #REQUIRED>
<!ATTLIST HTMLAttributeDefinition meta-qualifier CDATA #IMPLIED>
<!ATTLIST HTMLAttributeDefinition type NUMBER #REQUIRED>

<!ELEMENT XMLFieldDefinition EMPTY>
<!ATTLIST XMLFieldDefinition name CDATA #REQUIRED>
<!ATTLIST XMLFieldDefinition locator CDATA #REQUIRED>
<!ATTLIST XMLFieldDefinition ignore (YES|NO) NO>
<!ATTLIST XMLFieldDefinition priority CDATA #IMPLIED>
<!ATTLIST XMLFieldDefinition exclude (YES|NO) NO>

<!ELEMENT XMLAttributeDefinition EMPTY>
<!ATTLIST XMLAttributeDefinition name CDATA #REQUIRED>
<!ATTLIST XMLAttributeDefinition locator CDATA #REQUIRED>
```

```
<!ATTLIST XMLAttributeDefinition ignore (YES|NO) NO>
<!ATTLIST XMLAttributeDefinition priority CDATA #IMPLIED>
<!ATTLIST XMLAttributeDefinition type NUMBER #REQUIRED>
```

Semantik von Querverweisausdrücken (XPath)

Ein XPath-Ausdruck muss in Bezug auf einen Kontextknoten interpretiert werden und bezeichnet eine Gruppe von Knoten. Als Net Search Extender-Selektormuster verwendet ist der Kontextknoten frei, das heißt, ein relatives Pfadmuster *p* (für 'Pattern') wird als `//p` interpretiert.

Entsprechend dem XML-Datenmodell sind XML-Dokumente als Baumstrukturen mit folgenden Arten von Knoten zu betrachten:

- Der Stammknoten
- Elementknoten
- Textknoten
- Attributknoten
- Namensbereichsknoten
- Verarbeitungsanweisungsknoten
- Kommentarknoten

Die Verbindungen zwischen diesen Knoten, d. h. die strukturbildenden Beziehungen, geben die unmittelbare Einschlussbeziehung im XML-Dokument wieder.

Der *Stammknoten* kann nur am Ausgangspunkt und nirgends sonst in der Baumstruktur auftreten. Er enthält als untergeordnete Elemente das Dokumentelement und optionale Kommentare und Verarbeitungsanweisungen.

Elementknoten können beliebige Arten von Knoten außer dem Stammknoten enthalten. Die anderen Arten von Knoten sind nur als Endpunktknoten in der Baumstruktur zulässig.

Es gibt drei Arten von *Einschlussverbindungen*: untergeordnetes Element ('child'), Attribut ('attribute') und Namensbereich ('namespace'). Die Einschlussverbindungen 'attribute' und 'namespace' müssen zu Attribut- bzw. Namensbereichsknoten führen. D. h., um auf die untergeordneten Elemente eines Elementknotens (bei Diagrammen) zuzugreifen, müssen Sie den Attributverbindungen ('attribute') folgen, um alle enthaltenen Attribute zu finden, den Namensbereichsverbindungen ('namespace') folgen, um alle enthaltenen Namensbereichsdeklarationen zu finden, und den untergeordneten Verbindungen ('child') folgen, um übergeordnete Elemente, Textknoten, Verarbeitungsanweisungen und Kommentare zu finden.

Ein XPath-Ausdruck muss in Bezug auf einen Kontextknoten interpretiert werden und bezeichnet eine Gruppe von Knoten. Als Net Search Extender-Selektormuster verwendet ist der Kontextknoten frei, das heißt, ein relatives Pfadmuster *p* (für 'Pattern') wird als `//p` interpretiert.

Die XPath-Selektormuster für Net Search Extender sind wie folgt definiert:

- Pattern '| ' LocationPathPattern im Kontext N bezeichnet die Vereinigungsmenge der Knoten, die den Mustern Pattern und LocationPathPattern, beide im Kontext N, entsprechen.
- '/' RelativePathPattern im Kontext N bezeichnet alle Knoten, die dem Muster RelativePathPattern im Kontext des Stamms entsprechen.

- `'//'` `RelativePathPattern` im Kontext N bezeichnet die Vereinigungsmenge der durch das Muster `RelativePathPattern` bezeichneten Knoten in einem beliebigen Kontext, der ein untergeordneter Knoten (auf der 'Kindachse') des Stamms ist.
- `RelativePathPattern '/'` `StepPattern` entspricht einem Knoten im Kontext N, wenn und nur wenn dieser Knoten dem Muster `StepPattern` im Kontext des direkt übergeordneten Knotens entspricht, und der direkt übergeordnete Knoten dem Muster `RelativePathPattern` im Kontext N entspricht.
- `RelativePathPattern '//'` `StepPattern` entspricht einem Knoten im Kontext N, wenn und nur wenn dieser Knoten dem Muster `StepPattern` im Kontext des übergeordneten Knotens entspricht und es einen (nicht unbedingt direkt) übergeordneten Knoten gibt, der dem Muster `RelativePathPattern` im Kontext N entspricht.
- `'child'::NodeTest` (abgekürzte Syntax: `NodeTest`) im Kontext N entspricht einem Knoten, der ein direkt untergeordneter Knoten ('child') von N (auf der Kindachse) ist und der `NodeTest` erfüllt.
- `'attribute'::NodeTest` (abgekürzte Syntax: `@NodeTest`) im Kontext N entspricht einem Knoten, der ein Attribut von N ist und der `NodeTest` erfüllt.
- `NodeType '(' '')` ist für einen Knoten erfüllt, wenn und nur wenn er den angegebenen Typ besitzt.
- `'processing-instruction' '(' 'Literal '')` ist für jeden Knoten des Typs `Verarbeitungsanweisung` erfüllt, der den Namen `'Literal'` hat.
- `'*'` ist für jeden Element- oder Attributknoten erfüllt (Namensplatzhalterzeichen für Elementname).
- `NCName ':' '*'` ist für jeden Elementknoten erfüllt, der als Namenspräfix die Zeichenfolge `'NCName'` hat.
- `QName` ist für jeden Knoten erfüllt, der den angegebenen Namen hat.

Anmerkung

Ein Knotentest (`NodeTest`) der Form `'NameTest'` geht davon aus, dass der Knoten den Haupttyp der ausgewählten Achse hat. Das heißt, den Typ `'attribute'` auf der Attributachse und den Typ `'child'` auf der Kindachse. Aus diesem Grund kann `'NameTest'` nicht zur Auswahl von Kommentar- oder Verarbeitungsanweisungsknoten, sondern nur für Kind- und Attributknoten verwendet werden. Darüber hinaus lassen die Muster die Auswahl eines beliebigen Knotentyps außer Namensbereichsknoten zu, da die Achsenkennung `'namespace'` nicht zulässig ist.

Beispiele für Muster:

- `chapter | appendix` gibt alle Elemente `chapter` und Elemente `appendix` an.
- `table` bezeichnet alle Elemente `table`.
- `*` bezeichnet alle Elemente. (Beachten Sie, dass dies die Kurzform für `child::*` ist.)
- `ulist/item` gibt alle Elemente `item` an, die ein direkt übergeordnetes Element `ulist` haben.
- `appendix//subsection` bezeichnet alle Elemente `subsection` mit einem übergeordneten Element `appendix`.
- `/` bezeichnet die Einermenge, die nur den Stammknoten enthält.
- `comment()` gibt alle Kommentarknoten an.
- `processing-instruction()` gibt alle Verarbeitungsanweisungen an.
- `attribute::*` (oder `@*`) bezeichnet alle Attributknoten.

Die Syntax des Querverweiselements (Locator) ist wie folgt definiert:

```
Locator ::= LocationPathPattern
          | Locator '|' LocationPathPattern
LocationPathPattern ::= '/' RelativePathPattern ?
                    | '//'? RelativePathPattern
RelativePathPattern ::= StepPattern
                    | RelativePathPattern '/' StepPattern
                    | RelativePathPattern '//'? StepPattern
StepPattern ::= ChildOrAttributeAxisSpecifier NodeTest
ChildOrAttributeAxisSpecifier ::=
    ('child' | 'attribute') '::'
    | '@'?
NodeTest ::= NameTest
          | NodeType '(' ')'
          | 'processing-instruction' '(' Literal ')'
NameTest ::= '*' | NCName ':' '*' | QName
NodeType ::= 'comment' | 'processing-instruction'
```

NCName und QName entsprechen der Definition in der XML-Namensempfehlung (XML Names Recommendation).

NCName

Ein XML-Name, der keine Doppelpunkte enthält.

QName

Ein NCName, vor dem ein NCName, gefolgt von einem Doppelpunkt, stehen kann. Beispiel: NCName:NCName.

Begrenzungen für Textfelder und Dokumentattribute

Die Begrenzungen für Textfelder und Dokumentattribute sowie die in einem HTML-Dokumentmodell verwendbaren Tags sind jeweils in einer Liste zusammengefasst.

Im Folgenden werden die Begrenzungen für Textfelder und Dokumentattribute aufgeführt:

- Maximale Anzahl von Feldern in einem Index: 32767
- Maximale Anzahl von Werten für ein Attribut des Typs STRING in einem Dokument: 1024
- Maximale Anzahl von Attributen des Typs STRING: 253
- Die Anzahl von Zeichen in einem STRING-Attributwert wird auf 128 abgeschnitten.
- Maximale Anzahl von Attributen der Typen DATE und NUMBER: 32766
- Die Anzahl von Zeichen in einem DATE- oder NUMBER-Attribut wird auf 128 abgeschnitten.
- Für NUMBER-Attribute wird eine Gleitkommazahl mit doppelter Genauigkeit (DOUBLE) als Wert akzeptiert.
- Maximale Anzahl von Werten, die in einem Attribut des Typs DATE oder NUMBER in einem Dokument angegeben werden können: unbegrenzt

Die folgenden Tags können in einem HTML-Dokumentmodell verwendet werden:

- <A>
- <ADDRESS>
- <AU>
- <AUTHOR>
- <H1>

- <H2>, <H3>, <H4>, <H5>
- <H6>
- <TITLE>

Tags wie <HEAD> und <BODY>, die ihrerseits andere Tags enthalten, können in einem HTML-Dokumentmodell nicht als Textfeld angegeben werden.

Attributwerte für Outside In-Tag

Der folgende Abschnitt enthält eine Liste der möglichen Werte für das Tagattribut für Outside In-Dokumenteigenschaftstagtypen.

Gültige Werte für das Tagattribut für Outside In-Dokumenteigenschaft-Tagtypen:

SCCCA_ABSTRACT
 SCCCA_ACCOUNT
 SCCCA_ADDRESS
 SCCCA_ATTACHMENTS
 SCCCA_AUTHORIZATION
 SCCCA_BACKUPDATE
 SCCCA_BASEFILELOCATION
 SCCCA_BILLTO
 SCCCA_BLINDCOPY
 SCCCA_CARBONCOPY
 SCCCA_CATEGORY
 SCCCA_CHECKEDBY
 SCCCA_CLIENT
 SCCCA_COMPANY
 SCCCA_COMPLETEDDATE
 SCCCA_COUNTCHARS
 SCCCA_COUNTPAGES
 SCCCA_COUNTWORDS
 SCCCA_CREATIONDATE
 SCCCA_DEPARTMENT
 SCCCA_DESTINATION
 SCCCA_DISPOSITION
 SCCCA_DIVISION
 SCCCA_DOCCOMMENT
 SCCCA_DOCTYPE
 SCCCA_EDITMINUTES
 SCCCA_EDITOR
 SCCCA_FORWARDTO
 SCCCA_GROUP
 SCCCA_KEYWORD
 SCCCA_LANGUAGE
 SCCCA_LASTPRINTDATE
 SCCCA_LASTSAVEDBY
 SCCCA_MAILSTOP
 SCCCA_MANAGERSCCCA_MATTER
 SCCCA_OFFICE
 SCCCA_OPERATOR
 SCCCA_OWNER
 SCCCA_PRIMARYAUTHOR
 SCCCA_PROJECT
 SCCCA_PUBLISHER
 SCCCA_PURPOSE
 SCCCA_RECEIVEDFROM
 SCCCA_RECORDEDBY
 SCCCA_RECORDEDDATE
 SCCCA_REFERENCE
 SCCCA_REVISIONDATE
 SCCCA_REVISIONNOTES
 SCCCA_REVISIONNUMBER
 SCCCA_SECONDARYAUTHOR
 SCCCA_SECTION

SCCCA_SECURITY
SCCCA_SOURCE
SCCCA_STATUS
SCCCA_SUBJECT
SCCCA_TITLE
SCCCA_TYPIST
SCCCA_USERDEFINEDPROP
SCCCA_VERSIONDATE
SCCCA_VERSIONNOTES
SCCCA_VERSIONNUMBER

Gültige Werte für das Tagattribut für Outside-In-Starttag- und -Endtag-Subtypen:

SCCCA_ALTFONTDATA
SCCCA_ANNOTATIONREFERENCE
SCCCA_CAPTIONTEXT
SCCCA_CHARACTER
SCCCA_COMPILEDFIELD
SCCCA_COUNTERFORMAT
SCCCA_CUSTOMDATAFORMAT
SCCCA_DATEDEFINITION
SCCCA_DOCUMENTPROPERTYNAME
SCCCA_ENDNOTEREFERENCE
SCCCA_FONTANDGLYPHDATA
SCCCA_FOOTNOTEREFERENCE
SCCCA_FRAME
SCCCA_GENERATEDFIELD
SCCCA_GENERATOR
SCCCA_HYPERLINK
SCCCA_INDEX
SCCCA_INDEXENTRY
SCCCA_INLINEDATAFORMAT
SCCCA_LISTENTRY
SCCCA_MERGEENTRY
SCCCA_NAMEDCELLRANGE
SCCCA_REFERENCEDTEXT
SCCCA_STYLE
SCCCA_SUBDOCTEXT
SCCCA_TOA
SCCCA_TOAENTRY
SCCCA_TOC
SCCCA_TOCENTRY
SCCCA_TOF
SCCCA_VECTORSAVETAG
SCCCA_XREF

Beachten Sie, dass die Tabellen alle Dokumenteigenschaften sowie alle von den IN-SO-Filtern erkannten Tagsubtypen enthalten. Es gibt zwei Subtypausnahmen: SCCCA_DOCUMENTPROPERTY und SCCCA_BOOKMARK.

Teil 13. Referenz

Kapitel 55. Verwaltungsbefehle für den Instanzeigner

Zu den Verwaltungsaufgaben des Instanzeigners gehören die Überprüfung des Status der Sperren- und Aktualisierungsservices von Net Search Extender sowie das Starten und Stoppen dieser Services.

Dieser Abschnitt beschreibt die Syntax von Verwaltungsbefehlen für den Instanzeigner. Zu den Verwaltungsaufgaben des Instanzeigners gehören die Überprüfung des Status der Sperren- und Aktualisierungsservices von Net Search Extender sowie das Starten und Stoppen dieser Services.

Die Befehle sind Unterbefehle des Befehls **db2text**; sie ermöglichen die Verwaltung von Net Search Extender-Services, die für eine DB2-Instanz spezifisch sind.

Befehl	Zweck
„Befehl CONTROL“	Dient zum Auflisten und Löschen von Volltextindexsperrern. Listet außerdem die Cachestatus auf.
„Befehl START“ auf Seite 209	Startet die Net Search Extender-Instanzservices.
„Befehl STOP“ auf Seite 211	Stoppt die Net Search Extender-Instanzservices.
„DB2EXTHL, Befehl“ auf Seite 217	Ändert die maximale Größe des Eingabeparameters der benutzerdefinierten Hervorhebungsfunktion.

Befehl CONTROL

Dient zum Auflisten und Löschen von Volltextindexsperrern, die von den Net Search Extender-Instanzservices verwaltet werden.

Zweck

Wenn die Sperren- und Aktualisierungsservices aktiv sind, können Sie ihren Status und Informationen zum aktivierten Cache anzeigen.

In einer Umgebung mit partitionierten Datenbanken wirkt sich dieser Befehl nur auf die aktuelle Partition aus. Es liegt in der Verantwortung des Benutzers, den DB2-Befehl **db2_a11** für die erforderlichen Partitionen aufzurufen.

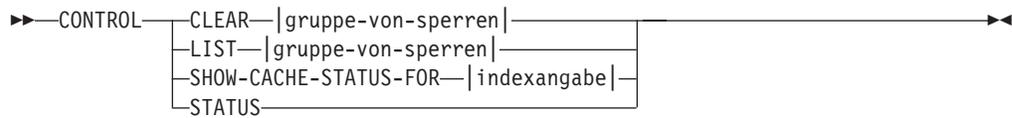
Berechtigung

Um diesen Befehl erfolgreich abzusetzen, muss der Benutzer der Eigner der DB2-Instanz sein und mit der Berechtigung DBADM und dem Zugriffsrecht DATAACCESS ausgestattet sein.

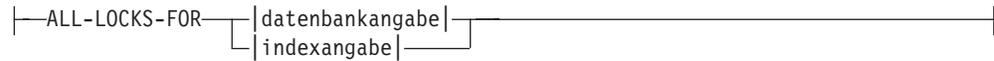
Erforderliche Verbindung

Dieser Befehl muss vom DB2-Datenbankserver aus ausgegeben werden.

Befehlssyntax



gruppe-von-sperren:



indexangabe:



datenbankangabe:



Befehlsparameter

CLEAR Verwenden Sie das Schlüsselwort **CLEAR**, um eine Bereinigung für eine Gruppe von Sperren zu erzwingen. Verwenden Sie diesen Befehl mit Vorsicht, nachdem Sie überprüft haben, was zu dem vorliegenden Sperrenproblem geführt hat.

Verwenden Sie den Befehl CLEAR nicht, solange noch Indexverwaltungsbefehle wie z. B. INDEX UPDATE für den Index aktiv sind, auf den Sie sie anwenden. Andernfalls könnte Ihr Index beschädigt und eine vollständige Wiederherstellung erforderlich werden.

LIST Verwenden Sie **LIST**, um Informationen über die aktuellen Sperren abzurufen, die für einen bestimmten Index oder eine Datenbank aktiviert sind. Ist eine Aktualisierungssperre vorhanden, werden mit dem Befehl auch Informationen über die Anzahl der bislang verarbeiteten Dokumente ausgegeben.

Hierbei ist zu beachten, dass dies nur für den Zeitraum gilt, in dem der Index gesperrt ist.

Wenn Sie eine Replikationserfassungstabelle verwenden, gibt es keine Aktualisierungsoperationen. Stattdessen können Einfügeoperationen entweder von einer Einfüge- oder einer Aktualisierungsoperation in der Quellentabelle, in der der Index erstellt wurde, ausgehen.

gruppe-von-sperren

Dient zur Eingrenzung auf die Sperren in der angegebenen Datenbank oder dem angegebenen Index.

SHOW CACHE STATUS FOR

Zeigt den Aktivierungsstatus für eine im Cache gespeicherte Tabelle des angegebenen Index. Die folgenden Status sind möglich: 'Nicht aktiviert' ('Not Activated') oder 'Zurzeit aktiviert' ('Currently Activated'). Wenn der Cache aktiviert ist, werden detaillierte Informationen zur Belegung des

Cachespeichers angezeigt. Hierzu gehören z. B. die maximale Cachegröße (in Megabyte), die maximale Anzahl einzufügender Dokumente und der in der Cachetabelle verbliebene Speicherbereich (in Kilobyte).

STATUS Wenn Sie das Schlüsselwort **STATUS** verwenden, zeigt der Befehl an, ob die Net Search Extender-Instanzservices zum Sperren und Aktualisieren betriebsbereit sind.

DATABASE *datenbankname*

Der Name der Datenbank auf dem Server, die verwendet wird.

INDEX *indexschema.indexname*

Das Schema und der Name des Textindex, der momentan verwendet wird. Dies entspricht der Angabe im Befehl **CREATE INDEX**.

Hinweise zur Verwendung

Wenn eine Fehlermeldung zu einem Verwaltungsbefehl anzeigt, dass ein Sperrenproblem vorliegt, stellen Sie sicher, dass keine sich gegenseitig behindernden Tasks aktiv sind. Dies kann z. B. der Fall sein, wenn versucht wird, einen Befehl **ALTER** auszuführen, während ein Befehl **UPDATE** ausgeführt wird.

Verwenden Sie **SHOW CACHE STATUS FOR** für eine Indexteilaktualisierung, um zu überprüfen, ob die angegebene Speichergröße immer noch ausreichend ist, um alle Aktualisierungsinformationen während der Indexaktualisierung aufzunehmen, oder um zu prüfen, ob eine Aktivierung ausgeführt wurde.

Befehl START

Startet einen Dämon, der das Sperren von Volltextindizes und die automatische Aktualisierung von Volltextindizes auf dem DB2-Server steuert.

Anmerkung: Da der Befehl keine temporär im Cache gespeicherte Tabelle für Indizes aktiviert, sind einzelne **ACTIVATE CACHE**-Befehle erforderlich, bevor Sie eine Suche mit einer gespeicherten Prozedur starten können.

Berechtigung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS für die aktuelle DB2-Instanz verfügen.

Erforderliche Verbindung

Dieser Befehl muss vom DB2-Datenbankserver aus ausgegeben werden.

Befehlssyntax

▶—START—◀

Befehlsparameter

Keine.

Hinweise zur Verwendung

Unter Windows startet der Befehl den Service DB2EXT - *instanzname* [-*knotennummer*]. Sie können den Service bzw. die Services auch über die Systemsteuerung oder mit dem Befehl **NET START** starten. Allerdings können Sie Net Search Extender nicht über einen Terminal-Service-Client starten.

Bei DB2-Instanzen, die mit partitionierten Datenbanken verwendet werden, empfiehlt es sich sehr, dass die Net Search Extender-Instanzservices mit dem Befehl **db2text start** anstatt mithilfe der normalen Windows-Methoden gestartet werden.

Mithilfe der Konsole für das Service-Management können Sie jeden der DB2EXT-Services für eine DB2-Instanz manuell starten oder stoppen. Um jedoch Net Search Extender in einem ordnungsgemäßen Laufstatus zu halten, ist es erforderlich, alle DB2EXT-Services zu starten und alle DB2EXT-Services zu beenden, die einer DB2-Instanz zugeordnet sind. Außerdem muss während des manuellen Startens und Stoppens unbedingt die folgende Reihenfolge eingehalten werden:

Starten

Starten Sie den NSE-Service (DB2EXT), indem Sie mit der Partition auf einem Host beginnen, die die kleinste Nummer aufweist, und arbeiten Sie sich zu der Partition mit der größten Nummer hoch, bevor Sie weitere DB2EXT-Services starten.

Stoppen

Stoppen Sie den NSE-Service (DB2EXT), indem Sie mit der Partition auf einem Host beginnen, die die größte Nummer aufweist, und arbeiten Sie sich abwärts zu der Partition mit der kleinsten Nummer durch.

Wenn Sie jedoch **db2text start** und **db2text stop** verwenden, wird diese erforderliche Abfolge vom System automatisch eingehalten.

Die Startart 'Automatisch' wird für die DB2EXT-Services nicht unterstützt. Die DB2EXT-Services müssen manuell gestartet werden, indem der Befehl **db2text start** oder der Befehl **net start** für jeden der DB2EXT-Services für die DB2-Instanz ausgeführt wird. Versuchen Sie nicht, Textsuchoperationen durchzuführen, wenn das System noch nicht vollständig gestartet wurde.

Wenn der Befehl **START** fehlschlägt, sind möglicherweise immer noch veraltete Einträge im Scheduler vorhanden, die auf nicht mehr vorhandene Indizes verweisen. Bearbeiten Sie die Datei `../sql1lib/db2ext/ctedem.dat`, und entfernen Sie sämtliche veralteten Einträge. Führen Sie den Befehl **START** erneut aus.

Nach dem erfolgreichen Start von Net Search Extender ist der Prozess **ctelock** (**ctelock.exe** unter Windows) auf Ihrem System aktiv. Es werden einige gemeinsam genutzte Ressourcen (gemeinsam genutzter Speicher und Semaphore) im Verzeichnis `/tmp` auf UNIX-Maschinen erstellt und gespeichert. Diese Dateien sind für Net Search Extender erforderlich und dürfen nicht gelöscht werden, solange die Instanz ausgeführt wird. Wenn allerdings nach dem Befehl **STOP** der Befehl **START** fehlschlägt, stellen Sie sicher, dass alle veralteten Ankerdateien im Verzeichnis `/tmp` entfernt wurden. Auf das Verzeichnis `/tmp` muss allgemeiner Lese-, Schreib- und Ausführungszugriff möglich sein.

Befehl STOP

Stoppt die Net Search Extender-Instanzservices zum Sperren und Aktualisieren.

Berechtigung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS für die aktuelle DB2-Instanz verfügen.

Erforderliche Verbindung

Dieser Befehl muss vom DB2-Datenbankserver aus ausgegeben werden.

Befehlssyntax

```
▶—STOP—┐
          └─FORCE─┘
```

Befehlsparameter

FORCE Stoppt Services, selbst wenn Prozesse Sperren aktiviert haben oder wenn die im Cache gespeicherte Tabelle für einen Index aktiviert ist. Wenn Sie **FORCE** nicht angeben, schlägt der Befehl in diesen Fällen mit einer Warnung für aktive Caches fehl.

Hinweise zur Verwendung

Wenn die Net Search Extender-Instanzservices gestoppt wurden, können bestimmte Net Search Extender-Befehle nicht mehr verwendet werden. Beim Neustart der Services müssen Sie den temporären Cachespeicher erneut aktivieren, wenn zuvor mit dem Index ein aktivierter Cache eingesetzt wurde.

Bei DB2-Instanzen, die mit partitionierten Datenbanken verwendet werden, empfiehlt es sich sehr, dass die Net Search Extender-Instanzservices mit dem Befehl **db2text stop** anstatt mithilfe der normalen Windows-Methoden gestoppt werden.

Mithilfe der Konsole für das Service-Management können Sie jeden der DB2EXT-Services für eine DB2-Instanz manuell stoppen. Um jedoch NSE in einem ordnungsgemäßen Laufstatus zu halten, ist es erforderlich, alle DB2EXT-Services zu beenden, die einer DB2-Instanz zugeordnet sind. Auch ist es während eines manuellen Stoppens erforderlich, dass Sie den NSE-Service (DB2EXT) stoppen, indem Sie mit der Partition auf dem Host beginnen, die die größte Nummer aufweist und sich abwärts zu der Partition mit der kleinsten Nummer durcharbeiten. Wenn Sie jedoch den Befehl **db2text stop** verwenden, wird diese erforderliche Abfolge vom System automatisch eingehalten.

Nach dem erfolgreichen Stoppen von Net Search Extender ist der Prozess **ctelock** (**ctelock.exe** unter Windows) beendet. Die gemeinsam genutzten Ressourcen und die Ankerdateien wurden aus dem Verzeichnis /tmp auf UNIX-Maschinen gelöscht.

Kapitel 56. Verwaltungsbefehle für den Datenbankadministrator

Der Datenbankadministrator kann verschiedene Befehle ausführen, um Datenbanken zur Verwendung mit Net Search Extender einzurichten.

Dieser Abschnitt beschreibt die Syntax von Verwaltungsbefehlen für den Datenbankadministrator. Die Datenbankverwaltung umfasst die Einrichtung von Datenbanken zur Verwendung durch Net Search Extender sowie die Inaktivierung dieser Einrichtung.

Nur die Befehle **ENABLE DATABASE** und **DISABLE DATABASE** sind Variationen des Befehls **DB2TEXT**, obwohl alle diese Befehle eine Verwaltung auf Datenbankebene ermöglichen.

Befehl	Zweck
„Befehl ENABLE DATABASE“	Aktiviert die aktuelle Datenbank zur Erstellung von Volltextindizes.
„Befehl DISABLE DATABASE“ auf Seite 215	Setzt die von Net Search Extender für eine Datenbank durchgeführten Vorbereitungsmaßnahmen zurück.
„DB2EXTHL, Befehl“ auf Seite 217	Mit dem Befehl DB2EXTHL wird die maximale Größe des Eingabeparameters der benutzerdefinierten Hervorhebungsfunktion geändert.

Tipp

Wenn im Befehl **db2text** keine Informationen zur Datenbankverbindung angegeben wurden, veranlasst die ausführbare Funktion **db2text** die Herstellung einer impliziten Verbindung zu dem Standard-Subsystem, das in der Umgebungsvariablen **DB2DBDFT** angegeben wurde.

Befehl ENABLE DATABASE

Aktiviert eine Datenbank, sodass Volltextindizes für Textspalten erstellt und verwendet werden können. Mit dem Befehl **ENABLE DATABASE** wird die Net Search Extender-Infrastruktur mit ihren Verwaltungstabellen, Sichten, benutzerdefinierten Funktionen (UDFs) und gespeicherten Prozeduren zum Suchen in der Datenbank erstellt.

Berechtigung

Ein Benutzer muss über die Berechtigung **DBADM** verfügen, um den Befehl **ENABLE DATABASE** ausführen zu können.

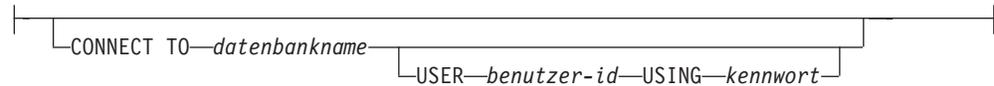
Voraussetzung

Der Instanzeigner muss über die Berechtigung **DBADM** und das Zugriffsrecht **DATAACCESS** verfügen. In Version 9.7 verfügt der **SYSADM** nicht mehr über die Berechtigung **SECADM** oder **DBADM**. Der **SECADM** muss dem Instanzeigner die Berechtigung **DBADM** und das Zugriffsrecht **DATAACCESS** explizit erteilen, bevor der Befehl **ENABLE DATABASE** ausgeführt werden kann.

Befehlssyntax



verbindungsoptionen:



Befehlsparameter

CONNECT TO *datenbankname*

Der Name der Datenbank, die das Ziel für diesen Befehl darstellt. Dieser Parameter kann ausgelassen werden, wenn die Umgebungsvariable **DB2DBDFT** definiert ist und der Benutzer den Befehl unter einer Benutzer-ID mit den erforderlichen DB2-Berechtigungen ausführt.

USER *benutzer-id* USING *kennwort*

Verwenden Sie eine *benutzer-id* und ein *kennwort* zur Herstellung einer Verbindung zur Datenbank.

AUTOGRANT

Wenn diese Option angegeben wird, wird versucht, dem Instanzeigner das Zugriffsrecht DBADM mit DATAACCESS zu erteilen, falls der Instanzeigner nicht über diese Zugriffsrechte für diese Datenbank verfügt. Damit das Erteilen der Zugriffsrechte erfolgreich ist, muss der Benutzer über das Zugriffsrecht SECADM für die Datenbank verfügen und darf nicht der Instanzeigner sein (ein Benutzer kann sich selbst keine Zugriffsrechte erteilen).

Anmerkung: Diese Option wird in der Schnittstelle für gespeicherte Prozeduren nicht unterstützt.

Hinweise zur Verwendung

Dieser Befehl bereitet die verbundene Datenbank für die Verwendung durch Net Search Extender vor. Die Ausführung dieses Befehls ist ein obligatorischer Schritt, damit Sie einen Net Search Extender-Index für Tabellen/Spalten in der Datenbank erstellen können.

Die Datenbankstandardwerte, die mit der Ausführung dieses Befehls festgelegt werden, können Sie der Katalogsicht DB2EXT.DBDEFAULTS entnehmen.

Änderungen an der Datenbank

Dieser Befehl erteilt dem DB2-Instanzeigner, dem die DB2-Instanz der aktivierten Datenbank zugeordnet ist, die Berechtigung DBADM.

Der Befehl **ENABLE DATABASE** erstellt verschiedene Datenbankobjekte im Schema DB2EXT, wie zum Beispiel Net Search Extender-Kataloge, benutzerdefinierte Funktionen (UDFs) und gespeicherte Prozeduren. Nach der Ausführung dieses Befehls sind die folgenden Katalogsichten verfügbar:

```
db2ext.dbdefaults
db2ext.textindexes
db2ext.textindexformats
db2ext.indexconfiguration
```

Diese Tabellen befinden sich im Standardtabellenbereich der Datenbank: IBMDEFAULTGROUP. Dieser Tabellenbereich ist über alle Knoten verteilt, die in der Datei db2nodes.cfg definiert sind.

Änderungen am Dateisystem

Keine.

Befehl DISABLE DATABASE

Macht die Änderungen rückgängig, die von Net Search Extender an einer Datenbank durchgeführt wurden.

Berechtigung

Ein Benutzer muss über die Berechtigung DBADM verfügen, um den Befehl **DISABLE DATABASE** ausführen zu können.

Voraussetzung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS verfügen. In Version 9.7 verfügt der SYSADM nicht mehr über die Berechtigung SECADM oder DBADM. Der SECADM muss dem Instanzeigner die Berechtigung DBADM und das Zugriffsrecht DATAACCESS explizit erteilen, bevor der Befehl **DISABLE DATABASE** ausgeführt werden kann.

Befehlssyntax

►►—DISABLE DATABASE FOR TEXT—┬—FORCE—┬—|verbindungsoptionen|—►►

verbindungsoptionen:

┌—CONNECT TO—datenbankname—┬—USER—benutzer-id—USING—kennwort—┐

Befehlsparameter

CONNECT TO *datenbankname*

Der Name der Datenbank, die das Ziel für diesen Befehl darstellt. Dieser Parameter kann ausgelassen werden, wenn die Umgebungsvariable DB2DBDFT definiert ist und der Benutzer den Befehl unter einer Benutzer-ID mit den erforderlichen DB2-Berechtigungen ausführt.

USER *benutzer-id* USING *kennwort*

Verwenden Sie eine *benutzer-id* und ein *kennwort* zur Herstellung einer Verbindung zur Datenbank.

FORCE

Erzwingt das Löschen aller Net Search Extender-Indizes in der Datenbank.

Hinweise zur Verwendung

Mit diesem Befehl wird die verbundene Datenbank zurückgesetzt, sodass sie nicht mehr von anderen Net Search Extender-Befehlen verwendet werden kann. Wenn die Datenbank Volltextindizes enthält, schlägt der Befehl fehl, sofern nicht die Option **FORCE** angegeben wird.

Dieser Befehl entzieht dem DB2-Instanzeigner nicht die Berechtigung DBADM.

Anmerkung: Das Inaktivieren einer Datenbank schlägt fehl, wenn in der Datenbank Textindizes definiert sind. Sie sollten diese Indizes einzeln entfernen und dann prüfen, ob irgendwelche Fehler auftreten. Wenn Sie den Befehl 'disable database for text force' verwenden, wird nur garantiert, dass Net Search Extender-Katalogtabellen in der Datenbank entfernt werden.

Können einige der Indizes jedoch nicht vollständig gelöscht werden, sind möglicherweise immer noch Ressourcen vorhanden, die manuell bereinigt werden müssen. Hierzu gehören z. B. die folgenden Komponenten:

- Dateien im Index-, Arbeits- und Cacheverzeichnis.
- Schedulereinträge in der Datei ctedem.dat.
- Wenn ein Index mit der Replikationserfassungsoption erstellt wurde, müssen die Einträge IBMSNAP_SIGNAL, IBMSNAP_PRUNE_SET und IBMSNAP_PRUNCNTL in den Tabellen der fernen Datenbank manuell gelöscht werden. Diese Einträge können mit dem Befehl APPLY_QUAL="NSE" || <instanzname> und TARGET_SERVER= *datenbankname* auf einfache Weise identifiziert werden.

Im folgenden Beispiel wird als Instanz DB2 und als Datenbank SAMPLE verwendet.

```
DELETE FROM <ccSchema>.IBMSNAP_SIGNAL
WHERE SIGNAL_INPUT_IN IN
      (SELECT MAP_ID FROM <ccSchema>.IBMSNAP_PRUNCNTL
       WHERE APPLY_QUAL= 'NSEDB2' AND TARGET_SERVER= 'SAMPLE');
```

```
DELETE FROM <ccSchema>.IBMSNAP_PRUNCNTL
WHERE APPLY_QUAL= 'NSEDB2' AND TARGET_SERVER= 'SAMPLE';
```

```
DELETE FROM <ccSchema>.IBMSNAP_PRUNE_SET
WHERE APPLY_QUAL= 'NSEDB2' AND TARGET_SERVER= 'SAMPLE';
```

Änderungen an der Datenbank

Die folgenden Änderungen, die zur Aktivierung von Net Search Extender an der Datenbank durchgeführt wurden, werden gelöscht:

- Die Net Search Extender-Katalogansichten in der Datenbank
- Alle von Net Search Extender erstellten Datenbankobjekte

Änderungen am Dateisystem und am gemeinsamen Speicher

Bei Verwendung der Option **FORCE** werden die Indexdateien gelöscht.

Bei Verwendung der Option **FORCE** wird der Cache gelöscht, sofern ein Cache für Indizes aktiviert ist.

DB2EXTHL, Befehl

Ändert die maximale Größe des Eingabeparameters der benutzerdefinierten Hervorhebungsfunktion.

Zweck

Standardmäßig verwendet die benutzerdefinierte Hervorhebungsfunktion als Eingabe ein Dokument von maximal 100 KB Größe und gibt ein CLOB-Objekt mit 200 KB zurück. Abhängig von der Größe des größten Dokuments in der Datenbank können Sie den Eingabewert bis zu einem Maximum von 1 GB erhöhen.

Berechtigung

Um diesen Befehl erfolgreich abzusetzen, muss der Benutzer Eigner einer DB2-Instanz und mit der Berechtigung DBADM und dem Zugriffsrecht DATAACCESS ausgestattet sein.

Erforderliche Verbindung

Dieser Befehl muss vom DB2-Datenbankserver aus ausgegeben werden und erfordert die Umgebungsvariable **DB2DBDFT**.

Befehlssyntax

►►—db2exthl—*new-highlight-input-size*—◄◄

Befehlsparameter

neue_eingabegröße_hervorhebung

Die neue Ergebnisgröße der benutzerdefinierten Hervorhebungsfunktion in KB. Dieser Wert ist eine positive ganze Zahl < 1048576.

Kapitel 57. Verwaltungsbefehle für den Texttabelleneigner

Der Tabelleneigner kann Befehle zum Ändern von Textindizes in Tabellen ausführen.

Dieser Abschnitt beschreibt die Syntax von Verwaltungsbefehlen für den Texttabelleneigner.

Die Befehle sind Unterbefehle des Befehls **DB2TEXT**. Sie ermöglichen dem Eigner einer Tabelle die Erstellung und Bearbeitung von Volltextindizes für Spalten der Tabelle.

Befehl	Zweck
„Befehl ACTIVATE CACHE“	Aktiviert den Cache, sodass Suchoperationen über die gespeicherte Prozedur möglich sind.
„Befehl ALTER INDEX“ auf Seite 221	Ändert die Merkmale eines Index.
„Befehl CLEAR EVENTS“ auf Seite 225	Löscht Indexereignisse aus einer Indexereignistabelle, die während der Indexaktualisierung verwendet wird.
„Befehl CREATE INDEX“ auf Seite 227	Erstellt einen Volltextindex.
„Befehl DEACTIVATE CACHE“ auf Seite 242	Inaktiviert den Cache, sodass Suchoperationen über die gespeicherte Prozedur nicht mehr möglich sind.
„Befehl DB2EXTTH“ auf Seite 245	Kompiliert die Thesaurusdefinitionsdatei.
„Befehl DROP INDEX“ auf Seite 244	Löscht einen Volltextindex für eine Textspalte.
„Befehl RESET PENDING“ auf Seite 246	Aktualisiert die textverwalteten abhängigen Tabellen für Basistabellen mit der erweiterten Zwischenspeicherinfrastruktur nach der Ausführung des Befehls SET INTEGRITY.
„Befehl UPDATE INDEX“ auf Seite 247	Startet den Indexierungsprozess auf der Basis des aktuellen Inhalts der Textspalten.
„Befehl HELP“ auf Seite 251	Zeigt die Liste der Optionen für den Befehl DB2TEXT an.
„Befehl COPYRIGHT“ auf Seite 252	Zeigt die Produkt- und Copyrightinformationen für Net Search Extender an.

Tipp:

Wenn im Befehl **db2text** keine Informationen zur Datenbankverbindung angegeben wurden, veranlasst die ausführbare Funktion **db2text** die Herstellung einer impliziten Verbindung zu dem Standard-Subsystem, das in der Umgebungsvariablen **DB2DBDFT** angegeben wurde.

Befehl ACTIVATE CACHE

Aktiviert die im Cache gespeicherte Tabelle entweder aus der DB2-Benutzertabelle oder aus den persistenten CACHEDATEIEN. Nach der Ausführung dieses Befehls sind Suchoperationen über eine gespeicherten Prozedur möglich.

Dieser Befehl ist nur verfügbar, wenn der Index mit der Option **CACHE TABLE** erstellt wurde.

Berechtigung

Die Berechtigungs-ID der Anweisung muss mindestens eine der folgenden Berechtigungen umfassen:

- Das Zugriffsrecht CONTROL für die Tabelle oder den Kurznamen, für die bzw. den der Textindex definiert wurde.
- Die Berechtigung DBADM.

Voraussetzung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS verfügen. In Version 9.7 verfügt der SYSADM nicht mehr über die Berechtigung SECADM oder DBADM. Der SECADM muss dem Instanzeigner die Berechtigung DBADM und das Zugriffsrecht DATAACCESS explizit erteilen, bevor der Befehl **ACTIVATE CACHE** ausgeführt werden kann.

Befehlssyntax

```
▶▶—ACTIVATE CACHE FOR INDEX—indexschema-"."—indexname—FOR TEXT—▶▶  
▶▶—RECREATE—|verbindungsoptionen|—▶▶
```

verbindungsoptionen:

```
|—CONNECT TO—datenbankname—USER—benutzer-id—USING—kennwort—|
```

Befehlsparameter

indexschema

Das Schema des Textindex, wie es im Befehl **CREATE INDEX** angegeben wurde. Wenn kein Schema angegeben wurde, wird die Benutzer-ID der DB2-Verbindung verwendet.

indexname

Der Name des Textindex, wie er im Befehl **CREATE INDEX** angegeben wurde.

RECREATE

Gilt nur für Indizes, die einen persistenten Cache verwenden. Ein vorhandener Cache wird gelöscht. Wenn eine Aktualisierung ohne Aktivierung ausgeführt wurde, wird der persistente Cache automatisch aus der Datenbank neu aufgebaut.

CONNECT TO *datenbankname*

Der Name der Datenbank, die das Ziel für diesen Befehl darstellt. Dieser Parameter kann ausgelassen werden, wenn die Umgebungsvariable **DB2DBDFT** definiert ist und der Benutzer den Befehl auf dem Server ausführt. Beachten Sie, dass die Benutzer-ID über die erforderlichen DB2-Berechtigungen verfügen muss.

USER *benutzer-id* **USING** *kennwort*

Verwenden Sie eine *benutzer-id* und ein *kennwort* zur Herstellung einer Verbindung zur Datenbank. Wenn Sie diese Angaben nicht machen, wird versucht, mit der aktuellen Benutzer-ID ohne Kennwort eine Verbindung herzustellen.

Hinweise zur Verwendung

Sie können den Befehl nicht absetzen, wenn einer der folgenden Befehle momentan für den Index ausgeführt wird:

- **UPDATE INDEX**
- **ALTER INDEX**
- **DROP INDEX**
- **CLEAR EVENTS**
- **DEACTIVATE CACHE**

Anmerkung: Zur Aktivierung einer im Cache gespeicherten Tabelle kann es erforderlich sein, die Tabelle völlig neu zu erstellen, auch wenn ein persistenter Cache verwendet wurde. Dies ist der Fall, wenn eine Aktualisierungsoperation ausgeführt wurde, während der persistente Cache inaktiviert war.

Die Speicherkapazität für die Erstellung des Caches wird anhand der aktuellen Anzahl von Dokumenten und der Größe der Ergebnisspalten dynamisch berechnet. Verwenden Sie den Wert für **PCTFREE**, um die berechnete Mindestspeicherkapazität um den Faktor $100/(100-PCTFREE)$ zu erhöhen. Der Wert für **PCTFREE** wird im Befehl **CREATE INDEX** oder **ALTER INDEX** angegeben.

Dementsprechend beschreibt der **PCTFREE**-Wert den Prozentsatz des zugeordneten Cache, der für Einfügeoperationen reserviert wird, während der Cache aktiviert ist. Beachten Sie, dass für jeden Befehl **ACTIVATE CACHE** die tatsächliche Speichergröße erneut ermittelt wird.

Änderungen am Dateisystem

Dateien zur Implementierung des persistenten Caches werden erstellt.

Befehl ALTER INDEX

Mit dem Befehl **ALTER INDEX** können die Merkmale eines Volltextindex geändert werden.

Zweck

Dieser Befehl ändert die Merkmale eines Volltextindex, wie zum Beispiel die Aktualisierungs- und Speicheroptionen.

Berechtigung

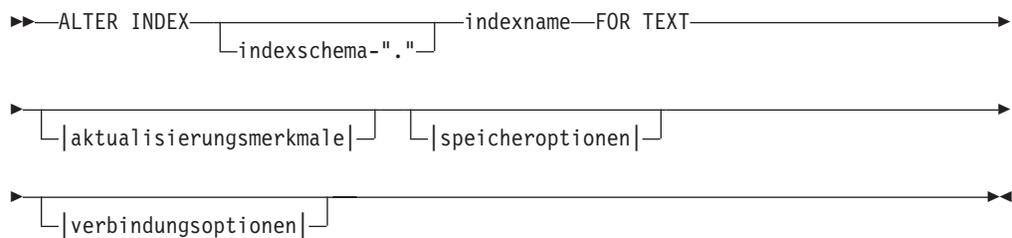
Die Berechtigungs-ID der Anweisung muss mindestens eine der folgenden Berechtigungen umfassen:

- Das Zugriffsrecht **CONTROL** für die Tabelle oder den Kurznamen, für die bzw. den der Textindex definiert wurde.
- Berechtigung **DBADM**

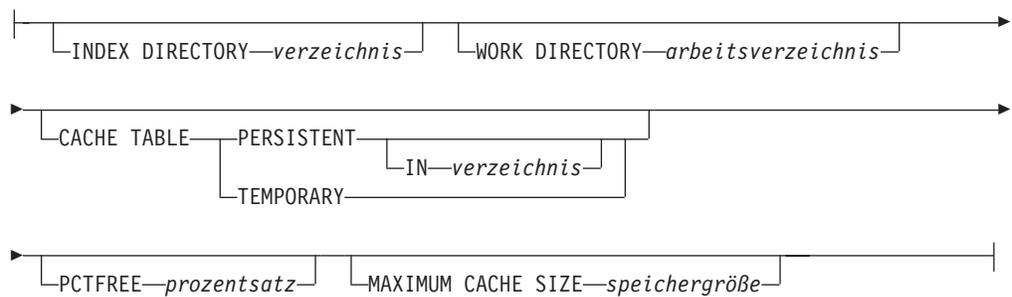
Voraussetzung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS verfügen. In Version 9.7 verfügt der SYSADM nicht mehr über die Berechtigung SECADM oder DBADM. Der SECADM muss dem Instanzeigner die Berechtigung DBADM und das Zugriffsrecht DATAACCESS explizit erteilen, bevor der Befehl **ALTER INDEX** ausgeführt werden kann.

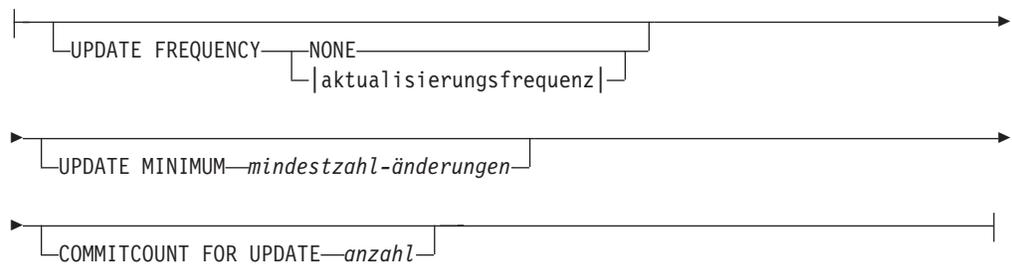
Befehlssyntax



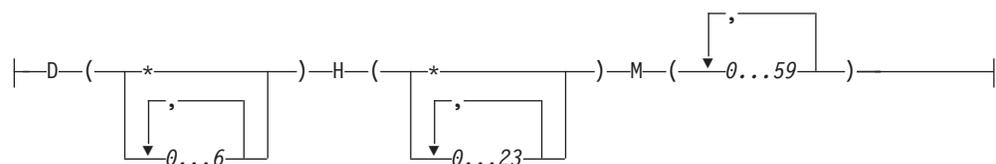
speicheroptionen:



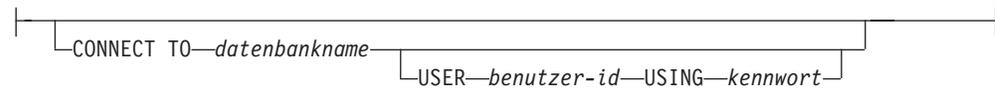
aktualisierungsmerkmale:



aktualisierungsfrequenz:



verbindungsoptionen:



Befehlsparameter

indexschema

Das Schema des Textindex, wie es im Befehl **CREATE INDEX** angegeben wurde. Wenn kein Schema angegeben wurde, wird die Benutzer-ID der DB2-Verbindung verwendet.

indexname

Der Name des Textindex, wie er im Befehl **CREATE INDEX** angegeben wurde.

INDEX DIRECTORY *verzeichnis*

Der Verzeichnispfad, in dem der Textindex gespeichert wird. Da das Verzeichnis Indexdaten enthält, müssen Sie sicherstellen, dass der Benutzer-ID des DB2-Instanzeigners die Zugriffsrechte zum Lesen, Schreiben und Ausführen für das Verzeichnis erteilt werden.

Beachten Sie, dass dieses Verzeichnis in einer Umgebung mit partitionierten Datenbanken in jeder Partition vorhanden sein muss. In diesem Verzeichnis wird das Unterverzeichnis `NODE<nr>` erstellt, um Indizes in logischen Partitionen eines Servers zu unterscheiden. Alle Indexdateien aus dem vorigen Indexverzeichnis werden gelöscht.

WORK DIRECTORY *arbeitsverzeichnis*

In diesem Verzeichnis werden temporäre Dateien während der Such- und Verwaltungsoperationen gespeichert. Sie können das separate Arbeitsverzeichnis unabhängig von einem neuen Indexverzeichnis ändern.

Wenn das Verzeichnis nicht vorhanden ist, wird es mit der Benutzer-ID des DB2-Instanzeigners erstellt. Wenn es vorhanden ist, müssen Sie sicherstellen, dass dem Instanzeigner auf UNIX-Plattformen die Zugriffsrechte zum Schreiben, Lesen und Ausführen für das Verzeichnis erteilt werden.

Beachten Sie, dass dieses Verzeichnis in einer Umgebung mit partitionierten Datenbanken in jeder Partition vorhanden sein muss. In diesem Verzeichnis wird das Unterverzeichnis `NODE<nr>` erstellt, um Indizes in logischen Partitionen eines Servers zu unterscheiden. Alle temporären Indexdateien aus dem vorigen Indexverzeichnis werden gelöscht.

CACHE TABLE PERSISTENT IN *verzeichnis*

Gibt an, dass die im Cache gespeicherte Tabelle **CREATE INDEX** auch nach einer Inaktivierung oder einem Warmstart des Systems persistent bleibt. In beiden Fällen ermöglicht dies eine schnelle Ausführung des Befehls **ACTIVATE CACHE**. Der persistente Cache wird im angegebenen Verzeichnis gespeichert.

Der zuvor erstellte persistente Cache wird an eine neue Position versetzt. Diese Operation setzt immer einen inaktiven Index voraus.

CACHE TABLE TEMPORARY

Gibt an, dass die im Cache gespeicherte Ergebnistabelle nun temporär ist und ein eventuell zuvor vorhandener persistenter Cache gelöscht wurde. Beachten Sie, dass für diese Änderungsoperation eine Inaktivierung des Index erforderlich ist.

MAXIMUM CACHE SIZE *speichergröße*

Gibt die neue maximale Größe der im Cache gespeicherten Tabelle an, die bei der Ausführung des Befehls **ACTIVATE CACHE** zu erstellen ist. Geben Sie den Parameter *speichergröße* in MB als positive ganze Zahl (INTEGER) an.

Wenn der Wert für *speichergröße* zu niedrig ist, schlägt der Befehl **ACTIVATE CACHE** fehl. Die tatsächliche Cachegröße wird bei der Ausführung des Befehls **ACTIVATE CACHE** berechnet. Diese Änderung setzt die Inaktivierung des Index voraus.

PCTFREE *prozentsatz*

Gibt den Prozentsatz des Caches an, der für weitere Dokumente freigehalten wird. Der Wert für *prozentsatz* muss als ganze Zahl kleiner als 100 und größer oder gleich 0 angegeben werden. Beachten Sie, dass der vorige persistente Cache gelöscht wird und dass diese Änderung die Inaktivierung des Index voraussetzt.

UPDATE FREQUENCY

Die Häufigkeit der Indexaktualisierung legt anhand der folgenden Parameter fest, wann die Aktualisierung durchgeführt wird:

- **D.** Die Wochentage, an denen der Index aktualisiert wird: * (jeden Tag) oder 0..6 (0=Sonntag)
- **H.** Die Stunden, zu denen der Index aktualisiert wird: * (jede Stunde) oder 0..23
- **M.** Die Minuten für die Indexaktualisierung: 0..59
- **NONE.** Es werden keine weiteren Indexaktualisierungen durchgeführt. Diese Angabe ist für eine Textspalte gedacht, an der keine weiteren Änderungen vorgenommen werden oder bei der künftig nur manuelle Aktualisierungen ausgeführt werden.

Wenn Sie das Schlüsselwort **UPDATE FREQUENCY** nicht angeben, bleiben die Häufigkeitseinstellungen unverändert.

UPDATE MINIMUM *mindestzahl-änderungen*

Die Mindestanzahl von Änderungen, die an Textdokumenten zulässig sind, bevor eine Teilaktualisierung des Index durchgeführt wird. Wenn Sie das Schlüsselwort **UPDATE MINIMUM** nicht angeben, bleibt die Einstellung unverändert.

Beachten Sie, dass Sie den Wert für das Schlüsselwort **UPDATE MINIMUM** nur ändern können, wenn Sie den Index nicht mit der Option **RECREATE ON UPDATE** erstellt haben.

COMMITCOUNT FOR UPDATE *anzahl*

Für die Aktualisierungsverarbeitung können Sie die Anzahl der Commits angeben. Dieser Wert gilt sowohl für den Befehl **UPDATE** als auch für die Angabe mit **UPDATE FREQUENCY**, durch die der Startzeitpunkt für die Aktualisierungsverarbeitung festgelegt wird.

Der Wert für **COMMITCOUNT** wird bei der Anfangsaktualisierung ignoriert.

Beachten Sie, dass Sie den Wert für das Schlüsselwort **COMMITCOUNT** nur ändern können, wenn Sie den Index nicht mit der Option **RECREATE ON UPDATE** erstellt haben.

Beachten Sie auch, dass Sie den Wert für das Schlüsselwort **COMMITCOUNT** nicht ändern können, wenn Sie den Index mit der Replikationsklausel (**REPLICATION**) erstellt haben.

CONNECT TO *datenbankname*

Der Name der Datenbank, die das Ziel für diesen Befehl darstellt. Dieser Parameter kann ausgelassen werden, wenn die Umgebungsvariable **DB2DBDFT** definiert ist und der Benutzer den Befehl auf dem Server ausführt. Beachten Sie, dass die Benutzer-ID über die erforderlichen DB2-Berechtigungen verfügen muss.

USER *benutzer-id* **USING** *kennwort*

Verwenden Sie eine *benutzer-id* und ein *kennwort* zur Herstellung einer Verbindung zur Datenbank. Wenn Sie diese Angaben nicht machen, wird versucht, mit der aktuellen Benutzer-ID ohne Kennwort eine Verbindung herzustellen.

Hinweise zur Verwendung

Sie können den Befehl zum Ändern des Index nicht absetzen, wenn einer der folgenden Befehle momentan für den Index ausgeführt wird:

- **ALTER INDEX**
- **CLEAR EVENTS**
- **ACTIVATE CACHE**
- **DROP INDEX**
- **UPDATE INDEX**
- **DEACTIVATE CACHE**

Wenn Sie den Index mit einer Cacheoption erstellen, können Sie den Befehl **ALTER INDEX** nicht für das Indexverzeichnis verwenden, wenn der Index aktiviert ist. Sie müssen den Cache zuerst inaktivieren.

In einer Umgebung mit partitionierten Datenbanken ist ein Textindex mit Cacheoptionen nur in Tabellenbereichen zulässig, die sich in einer einzigen Partition befinden.

Änderungen an der Datenbank

Die Net Search Extender-Katalogsichten werden geändert.

Änderungen am Dateisystem

- Im Index und in den Arbeitsverzeichnissen werden Unterverzeichnisse mit dem Namen **NODE<nr>** erstellt.
- Die Indexdateien werden verschoben.
- Es werden persistente Cacheverzeichnisse erstellt.
- Die persistenten Cachedateien werden verschoben.

Befehl CLEAR EVENTS

Löscht Indexierungsereignisse aus der Ereignissicht eines Index. Den Namen der Ereignissicht können Sie der Spalte **EVENTVIEWNAME** der Sicht **DB2EXT.TEXTINDEXES** entnehmen.

Berechtigung

Die Berechtigungs-ID der Anweisung muss mindestens eine der folgenden Berechtigungen umfassen:

- Das Zugriffsrecht **CONTROL** für die Tabelle oder den Kurznamen, für die bzw. den der Textindex definiert wurde.

- Die Berechtigung DBADM.

Voraussetzung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS verfügen. In Version 9.7 verfügt der SYSADM nicht mehr über die Berechtigung SECADM oder DBADM. Der SECADM muss dem Instanzeigner die Berechtigung DBADM und das Zugriffsrecht DATAACCESS explizit erteilen, bevor der Befehl **CLEAR EVENTS** ausgeführt werden kann.

Befehlssyntax

```

▶▶ CLEAR EVENTS FOR INDEX indexschema-".." indexname FOR TEXT

```

```

┌──────────┴──────────┐ ┌──────────┴──────────┐
COMMITCOUNT anzahl   | verbindungsoptionen |

```

verbindungsoptionen:

```

┌──────────┴──────────┐ ┌──────────┴──────────┐
CONNECT TO datenbankname | USER benutzer-id USING kennwort

```

Befehlsparameter

indexschema

Das Schema des Textindex, wie es im Befehl **CREATE INDEX** angegeben wurde. Wenn kein Schema angegeben wurde, wird die Benutzer-ID der DB2-Verbindung verwendet.

indexname

Der Name des Textindex, wie er im Befehl **CREATE INDEX** angegeben wurde.

COMMITCOUNT *anzahl*

Ein ganzzahliger Wert (INTEGER) ≥ 0 gibt die Anzahl der Zeilen an, die von DB2 in einer einzigen Transaktion gelöscht wurden.

CONNECT TO *datenbankname*

Der Name der Datenbank, die das Ziel für diesen Befehl darstellt. Dieser Parameter kann ausgelassen werden, wenn die Umgebungsvariable **DB2DBDFT** definiert ist und der Benutzer den Befehl auf dem Server ausführt. Beachten Sie, dass die Benutzer-ID über die erforderlichen DB2-Berechtigungen verfügen muss.

USER *benutzer-id* **USING** *kennwort*

Verwenden Sie eine *benutzer-id* und ein *kennwort* zur Herstellung einer Verbindung zur Datenbank. Wenn Sie diese Angaben nicht machen, wird versucht, mit der aktuellen Benutzer-ID ohne Kennwort eine Verbindung herzustellen.

Hinweise zur Verwendung

Wenn Sie mit der Option **UPDATE FREQUENCY** im Befehl **CREATE INDEX** oder **ALTER INDEX** regelmäßige Aktualisierungen definieren, sollten Sie regelmäßig die Ereignistabelle kontrollieren. Verwenden Sie **CLEAR EVENTS**, um die Ereignistabellen zu be-

reinigen, nachdem Sie die Ursache für sämtliche Ereignisse überprüft haben, die auf einen Fehler hinweisen, und die in der Ereignistabelle angegebene Fehlerquelle entfernt haben.

Sie können den Befehl zum Bereinigen von Ereignissen nicht absetzen, wenn einer der folgenden Befehle momentan für den Index ausgeführt wird:

- **UPDATE INDEX**
- **ALTER INDEX**
- **ACTIVATE CACHE**
- **DEACTIVATE CACHE**
- **DROP INDEX**

Befehl CREATE INDEX

Erstellt einen Volltextindex für eine Textspalte zur Verwendung durch Net Search Extender-Volltextabfragen.

Zweck

In einer Umgebung mit partitionierten Datenbanken wird ein Volltextindex in jeder Partition des Tabellenbereichs erstellt, in dem die Benutzertabelle definiert ist. Nachfolgende Änderungen an der Verteilung des Tabellenbereichs sind nicht zulässig und führen zu unvorhergesehenen Funktionsweisen der Verwaltungsbefehle sowie bei Suchoperationen.

Berechtigung

Die Berechtigungs-ID der Anweisung muss mindestens eine der folgenden Berechtigungen umfassen:

Eines aus dieser Gruppe:

- Berechtigung DBADM
- Das Zugriffsrecht CONTROL für die Tabelle oder den Kurznamen, für die bzw. den der Textindex definiert wurde.
- Das Zugriffsrecht INDEX für die Tabelle oder den Kurznamen, für die bzw. den der Textindex definiert wurde.

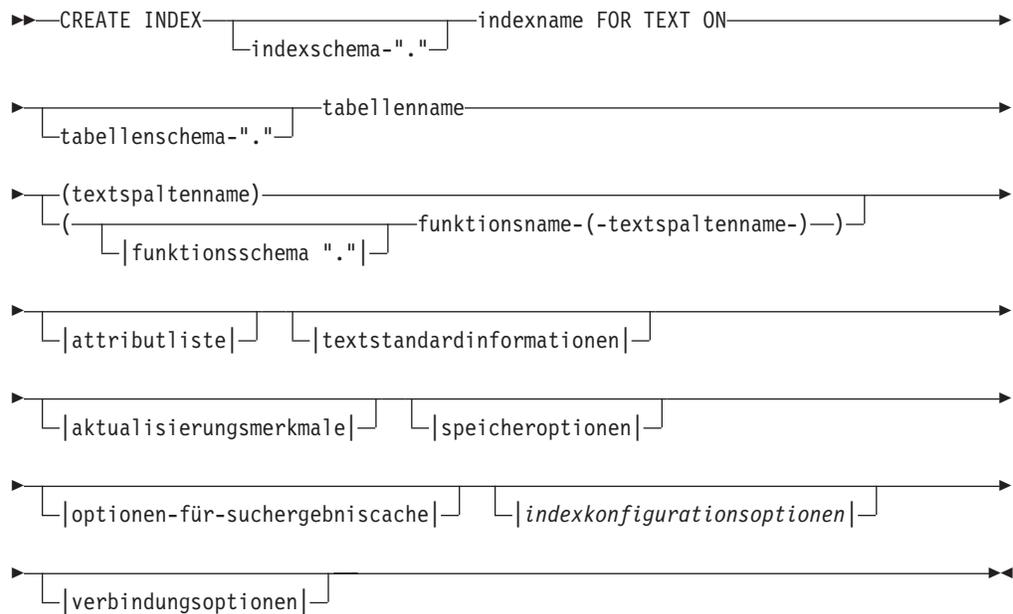
Sowie eines aus dieser Gruppe:

- Die Berechtigung IMPLICIT_SCHEMA für die Datenbank, wenn der implizite oder der explizite Schemaname des Index nicht vorhanden ist.
- Das Zugriffsrecht CREATEIN für das Schema, wenn der Schemaname des Index sich auf ein vorhandenes Schema bezieht.

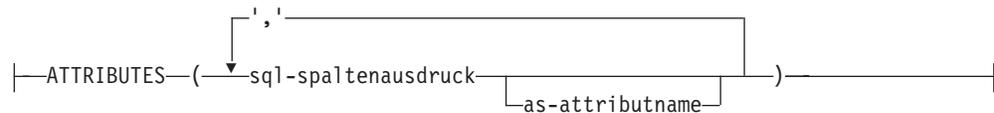
Voraussetzung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS verfügen. In Version 9.7 verfügt der SYSADM nicht mehr über die Berechtigung SECADM oder DBADM. Der SECADM muss dem Instanzeigner die Berechtigung DBADM und das Zugriffsrecht DATAACCESS explizit erteilen, bevor der Befehl **CREATE INDEX** ausgeführt werden kann.

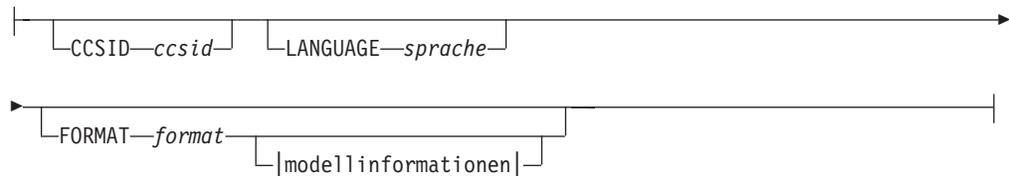
Befehlssyntax



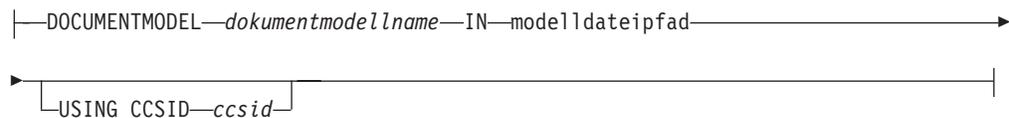
attributliste:



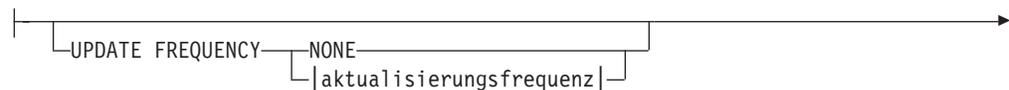
textstandardinformationen:

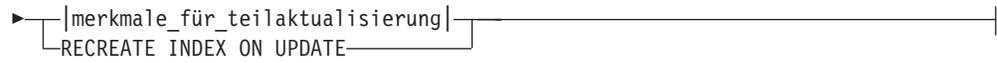


modellinformationen:

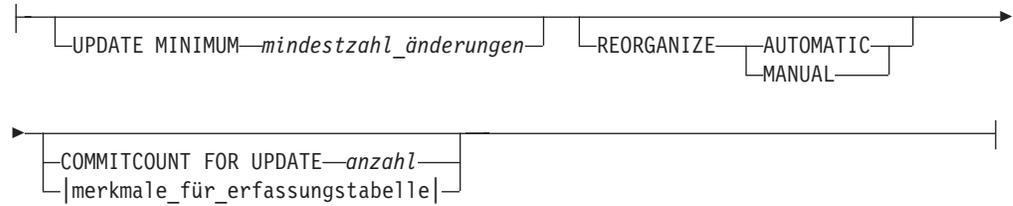


aktualisierungsmerkmale:

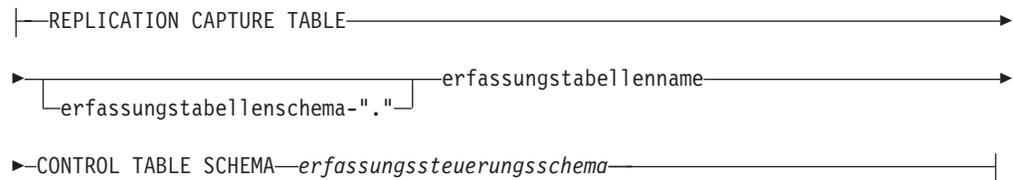




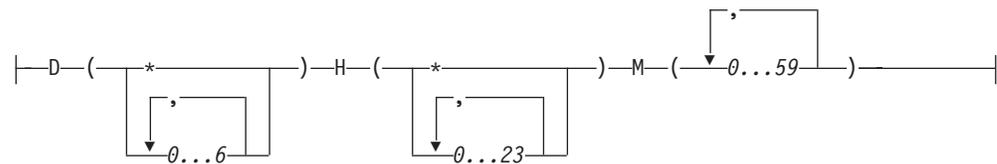
merkmale_für_teilaktualisierung:



merkmale_für_erfassungstabelle:



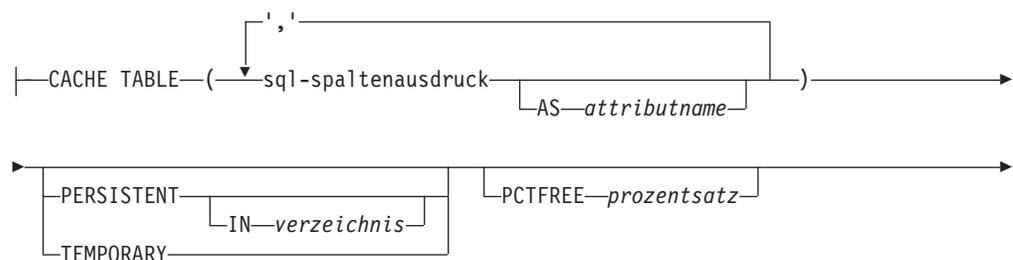
aktualisierungsfrequenz:

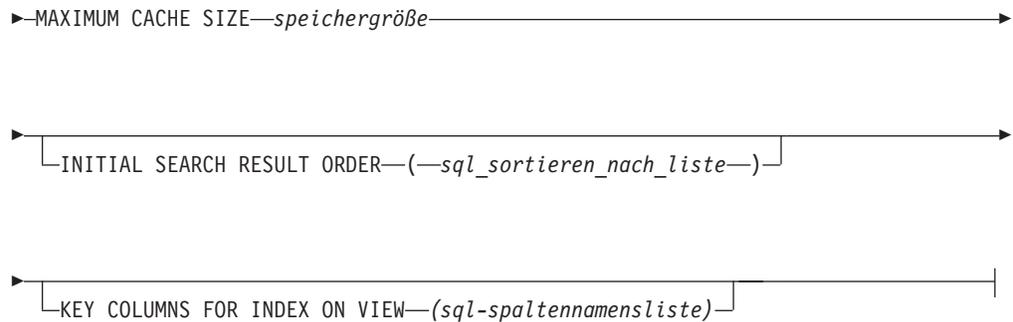


speicheroptionen:

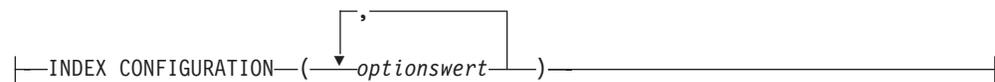


optionen-für-suchergebniscache:





indexkonfigurationsoptionen:



verbindungsoptionen:



Befehlsparameter

indexschema

Das Schema des Textindex. Verwenden Sie dieses Schema als DB2-Schemaname für die indexspezifischen Verwaltungstabellen. Wenn kein Schema angegeben wurde, wird die Benutzer-ID der DB2-Verbindung verwendet. Beachten Sie, dass das Indexschema ein gültiger DB2-Schemaname sein muss.

indexname

Der Name des Index. Zusammen mit dem Schemanamen dient dieser Name zur eindeutigen Identifizierung eines Volltextindex in einer Datenbank.

Beachten Sie, dass der Indexname ein gültiger DB2-Indexname sein muss.

tabellenschema

Das Schema der Tabelle, des Kurznamens oder der Sicht, für die der Index erstellt wird. Wenn kein Schema angegeben wurde, wird die Benutzer-ID der DB2-Verbindung verwendet.

tabellenname

Der Name der Texttabelle, des Kurznamens oder der Sicht in der verbundenen Datenbank, die die Spalte enthält, für die der Index erstellt werden soll.

Beachten Sie, dass die folgenden Einschränkungen zu berücksichtigen sind, wenn der Tabellenname keine DB2-Basistabelle bezeichnet:

- In einer Sicht können Sie nur eine Suche mit gespeicherten Prozeduren oder eine Suche mithilfe einer Tabellenwertfunktion ausführen. Daher müssen Sie mithilfe der Klausel **KEY COLUMNS FOR INDEX ON VIEW** die Schlüsselspalten für den Index oder die Sichten angeben.

- Bei Indexteilaktualisierungen für Kurznamen ohne Erfassungstabellen wird eine Protokolltabelle erstellt. Wenn die Daten in der Kurznamentabelle oder Sicht geändert werden, müssen Sie die Protokolltabelle manuell angeben. Bei Basistabellen wird die Protokolltabelle automatisch gefüllt; der Benutzer darf sie **NICHT** bearbeiten.
- Die DB2-Vergleichselemente CONTAINS, SCORE und NUMBEROFMATCHES sind nur für Basistabellen und Kurznamen, jedoch nicht für Sichten zulässig.
- Indizes oder Sichten sind nur zulässig, wenn Sie im Befehl Optionen für den Suchergebniscache angeben.
- Die erweiterte textverwaltete Zwischenspeicherinfrastruktur, die das auf Triggern beruhende Protokoll ergänzt, um Teilaktualisierungen zu unterstützen, kann auf partitionierte und nichtpartitionierte Basistabellen angewendet werden. Sie wird jedoch nicht für Sichten oder Kurznamen unterstützt.

textspaltenname

Der Name der Spalte, die den Text enthält, der zur Erstellung des Volltextindex verwendet wird. Die Spalte muss einen der folgenden Datentypen aufweisen:

- CHAR (FOR BIT DATA)
- VARCHAR (FOR BIT DATA)
- LONG VARCHAR (FOR BIT DATA)
- CLOB
- DBCLOB
- BLOB
- GRAPHIC
- VARGRAPHIC
- LONG VARGRAPHIC
- XML

Wenn die Spalte keinen dieser Datentypen aufweist, geben Sie eine Umsetzungsfunktion der Form *funktionsschema.funktionsname* an, um den Spaltentyp in einen unterstützten Spaltentyp umzuwandeln.

Beachten Sie, dass für Spalten mehrere Indizes zulässig sind, jedoch **nur**, wenn eine der folgenden Bedingungen gilt:

Der Index wird für eine Sicht erstellt.

Dies bedeutet, dass der Index nicht in den Suchargumenten CONTAINS, SCORE oder NUMBEROFMATCHES verwendet werden kann.

Der Index wird für eine Tabelle erstellt.

Wenn alle Indizes synchronisiert sind, stimmen die Eigenschaften für eine Spalte, die in den folgenden Details des Befehls **CREATE INDEX** festgelegt werden, bei allen Indizes überein:

- Name und Schema der Funktion
- ATTRIBUTES
- CCSID
- LANGUAGE
- FORMAT
- DOCUMENTMODEL
- INDEX CONFIGURATION

Daher spielt es in diesem Fall keine Rolle, welcher Index mit dem Argument CONTAINS, SCORE oder NUMBEROFMATCHES ausgewählt wird.

funktionsschema.funktionsname

Das Schema und der Name einer benutzerdefinierten Funktion, die für den Zugriff auf Textdokumente verwendet wird, die sich in einer Spalte mit einem nicht unterstützten Typ befinden. Die Funktion führt eine Spaltentypumwandlung durch; dabei wird als Eingabeparameter ein beliebiger Spaltentyp verwendet. Der Wert wird als ein von Net Search Extender unterstützter Typ zurückgegeben.

ATTRIBUTES (*sql-spaltenausdruck AS attributname, ...*)

Stellt sicher, dass zusätzlich zur Textspalte auch der Inhalt eines Spaltenausdrucks indexiert wird. Dieser Inhalt kann anschließend mit der Klausel **ATTRIBUTES** in einer Suchanweisung durchsucht werden. Die SQL-Spaltenausdrücke müssen mithilfe von Spaltennamen ohne Qualifikationsmerkmal aus der Tabelle definiert werden, auf deren Grundlage der Index erstellt wird. Es ist nur der Datentyp DOUBLE zulässig. In den Spaltenausdrücken können Umsetzungsoperatoren verwendet werden, eine implizite Typumsetzung von DB2 ist jedoch **nicht** möglich. Die Attributnamen müssen den Regeln für Attributnamen in Dokumentmodellen entsprechen und sich von allen Attributnamen in der Modelldefinitionsdatei für Indizes unterscheiden.

Ermitteln Sie die Attributnamen für Ausdrücke anhand der folgenden Regeln:

- Wenn die Namen in der SQL-Klausel **AS** im Spaltenausdruck explizit genannt werden, verwenden Sie den angegebenen Namen. Beispiel:
ATTRIBUTES (C1+C2 AS meinname)
- Wenn eine Spalte der angegebenen Tabelle ohne **AS** verwendet wird, wird der Name der Spalte verwendet. Beispiel:
ATTRIBUTES (C1)
- Wenn ein Ausdruck ohne **AS** verwendet wird und nicht auf eine angegebene Spalte verweist, meldet der Befehl **CREATE INDEX** einen Fehler.

Beispiel:

```
ATTRIBUTES (CAST(JULIAN_DAY(date) AS DOUBLE) as day, (price1+price2)/2 as avg_price)
```

Beachten Sie, dass Attribute, die nicht zwischen einfachen Anführungszeichen stehen, groß geschrieben werden und beim Suchvorgang in Großbuchstaben angegeben werden müssen.

CCSID *ccsid*

Die ID für den codierten Zeichensatz (CCSID = Coded Character Set Identifier) wird beim Indexieren von Textdokumenten verwendet. Der Standardwert wird der Sicht DB2EXT.DBDEFAULTS mit dem Argument DEFAULTNAME='CCSID' entnommen.

Setzen Sie nur dann eine CCSID, wenn der Datentyp der Spalte binär ist.

LANGUAGE *sprache*

Mit dem Sprachenparameter wird die Sprache des Stoppwörterverzeichnis angegeben, das ausgewählt wird, wenn der Wert für die Indexkonfiguration IndexStopWords auf 0 gesetzt wird (Stoppwörter bei der Indexierung ignorieren). Dieser Parameter muss für Thailändisch (TH_TH) immer gesetzt sein, damit der Umbruch für thailändische Wörter möglich ist und damit im Türkischen die Verwendung des 'i' mit und ohne Punkt richtig unterschieden wird.

FORMAT *format*

Das Format der Textdokumente in der Spalte, zum Beispiel HTML. Diese Information wird bei der Indexierung von Dokumenten benötigt.

Informationen für Formate strukturierter Dateien können Sie in einer Dokumentmodelldatei angeben. Wenn keine Dokumentmodelldatei angegeben ist, wird der Text des Dokuments mithilfe eines Standarddokumentmodells indexiert.

Wenn das Schlüsselwort **FORMAT** nicht angegeben wird, wird der Standardwert der Sicht `DB2EXT.DBDEFAULTS` mit dem Argument `DEFAULTNAME='FORMAT'` entnommen. Der von Net Search Extender gesetzte Anfangsstandardwert ist `TEXT`. Für den Datentyp XML ist die XML-Kennung **FORMAT** verbindlich.

DOCUMENTMODEL *dokumentmodellname* **IN** *modelldateipfad*

Der Parameter *modelldateipfad* gibt die Position einer Modelldatei an. Hierbei muss es sich um einen vollständig qualifizierten Pfad handeln. Die Modelldatei enthält eine Modelldefinition für das Format in der Klausel **FORMAT**. Der DB2-Instanzeigner muss Lesezugriff auf diese Datei haben. Mit einem Dokumentmodell können Sie bestimmte Abschnitte in einem Dokument indexieren und durchsuchen. Außerdem können Sie in einem Dokumentmodell Formatierungssteuerbefehle (Tags) und Abschnittsnamen definieren. Ein Dokumentmodell ist an ein Dokumentformat gebunden, das HTML-, XML- oder GPP-Strukturen unterstützt. In einer Modelldatei können Sie nur ein Dokumentmodell angeben.

Beachten Sie, dass alle späteren Änderungen im Index nicht berücksichtigt werden, da das Dokumentmodell nur bei der Ausführung des Befehls **CREATE INDEX** gelesen wird.

Beachten Sie, dass in einer Umgebung mit partitionierten Datenbanken ein gemeinsam genutztes Dateisystem verwendet werden muss, um sicherzustellen, dass der *modelldateipfad* auf allen Knoten unter Linux- oder UNIX-Betriebssystemen zugänglich ist. Unter Windows-Betriebssystemen muss dagegen für den *modelldateipfad* für Dokumentmodelldateien ein lokaler Pfad definiert werden, der auf jedem Knoten verfügbar ist.

USING CCSID *ccsid*

Dient zur Angabe der ID des codierten Zeichensatzes (CCSID), der zur Interpretation des Inhalts der Modelldatei verwendet wird. Der Standardwert wird der Sicht `DB2EXT.DBDEFAULTS` mit dem Argument `DEFAULTNAME='MODELCCSID'` entnommen.

UPDATE FREQUENCY

Die Häufigkeit der Indexaktualisierung legt fest, wann die Aktualisierung durchgeführt wird. Wenn weniger Änderungen an der Benutzertabelle durchgeführt wurden, als mit der Option **UPDATE MINIMUM** vorgegeben, wird der Index nicht aktualisiert. Wenn Sie **UPDATE FREQUENCY** nicht angeben, wird der Standardwert `NONE` verwendet, sodass keine Indexaktualisierungen durchgeführt werden. Dies ist nützlich, wenn keine weiteren Änderungen an einer Textspalte mehr geplant sind oder wenn Sie den Aktualisierungsprozess manuell steuern möchten.

- **D.** Die Wochentage, an denen der Index aktualisiert wird: * (jeden Tag) oder 0..6 (0=Sonntag)
- **H.** Die Stunden, zu denen der Index aktualisiert wird: * (jede Stunde) oder 0..23
- **M.** Die Minuten für die Indexaktualisierung: 0..59

- **NONE.** Es werden keine weiteren Indexaktualisierungen durchgeführt. Die Aktualisierung muss manuell gestartet werden.

Der Standardwert wird der Sicht DB2EXT.DBDEFAULTS mit dem Argument DEFAULTNAME='UPDATEFREQUENCY' entnommen.

Wenn Sie den Parameter **UPDATE FREQUENCY** zur Planung von automatischen Indexaktualisierungen nicht verwenden möchten, können Sie auch Betriebssystemfunktionen (z. B. crontab) verwenden.

UPDATE MINIMUM *mindestzahl_änderungen*

Die Mindestanzahl von Änderungen, die an Textdokumenten erforderlich sind, bevor der Index auf der Basis der **UPDATE FREQUENCY**-Einstellungen aktualisiert wird. Es sind nur positive ganzzahlige Werte (INTEGER) zulässig. Der Standardwert wird der Sicht DB2EXT.DBDEFAULTS mit dem Argument DEFAULTNAME='UPDATEMINIMUM' entnommen.

Beachten Sie, dass dieser Wert ignoriert wird, wenn der Befehl **DB2TEXT UPDATE** manuell ausgeführt wird. Diese Option kann nicht mit der Option **RECREATE INDEX ON UPDATE** verwendet werden, da die Anzahl von Änderungen nicht ohne eine Protokolldatei und Trigger für die Teilaktualisierung (inkrementelle Aktualisierung) verfügbar ist.

In Umgebungen mit partitionierten Datenbanken wird **UPDATE MINIMUM** in jeder Partition geprüft.

REORGANIZE AUTOMATIC | MANUAL

Aktualisierungen, die auf der Basis der Einstellungen zur Aktualisierungsfrequenz ausgeführt werden, organisieren den Index neu, wenn **REORGANIZE AUTOMATIC** angegeben wird. Dieser Schritt wird automatisch gemäß dem Wert durchgeführt, der mit `select REORG SUGGESTED from DB2EXT.TEXTINDEXES` nach der Aktualisierung abgerufen wird.

REORGANIZE MANUAL kann nur über einen manuellen **UPDATE**-Befehl mit der Option **REORGANIZE** ausgeführt werden.

Wenn die Klausel **REORGANIZE** nicht angegeben wird, wird der Standardwert der Sicht DB2EXT.DBDEFAULTS mit dem Argument DEFAULTNAME='AUTOMATICREORG' entnommen.

REPLICATION CAPTURE TABLE *erfassungstabellenschema.erfassungstabellenname* **CONTROL TABLE SCHEMA** *erfassungssteuerungsschema*

Bei der Verarbeitung von Teilaktualisierungen wird die angegebene Replikationserfassungstabelle verwendet und nicht eine Protokolltabelle oder eine textverwaltete Zwischenspeichertabelle, die sonst für den Index erstellt wird. Aus diesem Grund beziehen sich der Schemaname, der Tabellename und der Name der Replikationserfassungstabelle auf Objekte in der lokalen (förderierten) DB2-Datenbank.

Das Erfassungssteuerungsschema ist der Schemaname der Replikationssteuertabellen im lokalen DB2-System, z. B. IBMSNAP_PRUNE_SET. Die Replikationssteuertabellen müssen nach der Einrichtung der Replikation im lokalen DB2-System als Kurznamen verfügbar sein.

Es müssen mindestens Kurznamen für die folgenden Erfassungssteuertabellen vorhanden sein:

- IBMSNAP_SIGNAL
- IBMSNAP_PRUNE_SET
- IBMSNAP_PRUNCNTL
- IBMSNAP_REGISTER

- IBMSNAP_REG_SYNC (nur ferne Nicht-DB2-Datenbankquellen)

Da die DB2-Replikationszentrale nicht automatisch sicherstellt, dass lokale Kurznamen für eine ferne Erfassungstabelle und für Replikationssteuertabellen erstellt werden, müssen Sie diese Task möglicherweise manuell durchführen. Diese Task gleicht der Erstellung eines Kurznamens für die Tabelle, in der der Textindex erstellt werden soll.

Die Spaltennamen der Primärschlüsselspalten im Kurznamen der Benutzertabelle und im Kurznamen der Erfassungstabelle müssen übereinstimmen. Außerdem dürfen die Namen der Spalten IBMSNAP_OPERATION, IBMSNAP_COMMITSEQ und IBMSNAP_INTENTSEQ im Kurznamen der Erfassungstabelle nicht geändert werden.

Nach der Indexerstellung verweist sowohl der Spaltenname DB2EXT.TEXTINDEXES(LOGVIEWNAME) als auch der Spaltenname DB2EXT.TEXTINDEXES(LOGVIEWSCHEMA) auf den lokalen Namen der Replikationserfassungstabelle.

Da Net Search Extender nicht die gesamte Funktionalität der DB2-Replikationszentrale benötigt, müssen für die CD-Tabelle (CD = Change Data) bzw. die CCD-Tabelle (CCD = Consistent Change Data) die folgenden Regeln gelten:

- Verwenden Sie die Registrierung der Änderungserfassung und nicht die Kopieroption für die vollständige Aktualisierung.
- Eine horizontale Untergruppierung von Erfassungsänderungen ist nicht zulässig. Diese kann zum Beispiel mit Triggern ausgeführt werden. Siehe Kapitel 6 zur Erstellung von Untergruppen zu Daten in Ihrer Replikationsumgebung im *DB2 Replikation Referenzhandbuch*.
- Die Registrierung von Änderungen für eine Untergruppe von Spalten ist nur zulässig, wenn die Primärschlüsselspalten, die Textspalte und alle Spalten im Zusammenhang mit den Attribut- und Cachetabellenausdrücken des Befehls **DB2TEXT CREATE INDEX** berücksichtigt werden.
- Die Primärschlüsselspalten müssen in die Erfassungstabelle aufgenommen werden. Dabei sind die Nachimagespalten ausreichend.
- Die Erfassungstabellen dürfen nicht komprimiert werden. Für jeden Primärschlüssel muss ein Eintrag mit den aktuellen Daten existieren. Für Net Search Extender muss jedoch ein vollständiges Protokoll vorhanden sein.
- Die Tabelle muss die D/I-Option verwenden. Mit dieser Option können Aktualisierungen von Primärschlüsseln der Quellentabelle in Paare von INSERT- und DELETE-Operationen umgesetzt werden.

Informationen zu den unterstützten fernen Quellenversionen in DB2 Version 9.7 finden Sie auf den Seiten der technischen Unterstützungsfunktion.

Anmerkung:

Stellen Sie sicher, dass der korrekte Quellentabellenname in die Registrierungstabelle eingefügt wird. Abhängig vom Typ des fernen Datenbankverwaltungssystems (DBMS) muss der ferne Tabellenname oder der lokale Kurzname verwendet werden:

- DB2: ferner Tabellenname (d. h., der Tabellenname auf dem fernen Server)
- Andere Datenbanksysteme: lokaler Kurzname (d. h., der entsprechende Kurzname in der föderierten DB2-Datenbank)

Es ist eine Benutzerzuordnung erforderlich, die dem lokalen Benutzer den Zugriff auf die ferne Datenquelle mithilfe von Kurznamen ermöglicht. Der ferne Benutzer muss das Zugriffsrecht CONTROL für die Tabellen besitzen.

Wenn die Benutzer-ID des DB2-Instanzeigners von der lokalen Benutzer-ID abweicht, ist eine zusätzliche Benutzerzuordnung für die Benutzer-ID des DB2-Instanzeigners erforderlich.

Der angegebene Basistabellenname darf keine Sicht auf einen Kurznamen sein. Eine Sicht kann nämlich eine Sicht auf mehrere Kurznamen sein und außerdem können mehrere CD- und CCD-Tabellen beteiligt sein. Da in der Replikationserfassungsklausel nur eine CD- oder CCD-Tabelle angegeben werden kann, kann eine Sicht auf Kurznamen nicht unterstützt werden. Auch können Kurznamen in fernen Sichten nicht unterstützt werden, da der Primärschlüssel fehlt.

Die CD- bzw. CCD-Tabelle muss als Kurzname definiert sein. Die Verwendung einer Sicht oder eines Aliasnamens ist hingegen nicht zulässig.

Die Option **COMMITCOUNT** kann nicht verwendet werden, wenn die Option **REPLICATION CAPTURE TABLE** für den Befehl **CREATE INDEX** verwendet wird.

COMMITCOUNT FOR UPDATE *anzahl*

Für die **Teilaktualisierung** kann eine Anzahl von Commits angegeben werden. Falls diese Option nicht angegeben wird, wird der Standardwert der Sicht DB2EXT.DBDEFAULTS mit dem Argument DEFAULTNAME='COMMITCOUNT' entnommen.

Der Wert für **COMMITCOUNT FOR UPDATE** für den Index ist in DB2EXT.TEXTINDEXES.COMMITCOUNT zu finden. Dieser Wert kann für jeden Index mit dem Befehl **ALTER INDEX** geändert werden. Er gilt auch für die Aktualisierungsverarbeitung, die mithilfe der Angabe **UPDATE FREQUENCY** festgelegt wird. Der Wert 0 bedeutet, dass die Aktualisierung in einer Transaktion abgeschlossen wird. Werte größer 0 geben die Anzahl von Dokumenten an, die in einer Transaktion zu verarbeiten sind. Es ist nicht empfehlenswert, einen COMMITCOUNT ungleich null zu verwenden; falls dies jedoch unumgänglich ist, sollten Sie den Wert hoch genug setzen, damit die Anzahl unmittelbarer Commits bei einer einzigen Teilaktualisierung äußerste gering bleibt. Jeder Commit beinhaltet das Verschieben von Indexdateien und das Bereinigen der Indexprotokolldatei; hierfür ist eine beachtliche Menge an Zeit erforderlich, insbesondere dann, wenn es zu einer wiederholten Durchführung kommt. Der COMMITCOUNT (Commitzähler) wird übergreifend auf alle Änderungen für Dokumente angewendet, die in der Protokolltabelle als auch in der erweiterten textverwalteten Zwischenspeicherinfrastruktur aufgelistet sind, falls diese für den Index konfiguriert wurde.

Wenn COMMITCOUNT nicht definiert wurde, wird der Parameter NUMBER_DOCS aus db2ext.textindexes nicht aktualisiert. Wenn Sie die Anzahl der bereits verarbeiteten Dokumente während des Aktualisierungsprozesses anzeigen wollen, müssen Sie deshalb den Befehl CONTROL LIST verwenden.

RECREATE INDEX ON UPDATE

Diese Option lässt keine inkrementellen Aktualisierungen (Teilaktualisierungen) von Indizes zu, sondern erstellt den Index erneut, wenn eine Aktualisierung (durch einen UPDATE-Befehl oder durch eine regelmäßige Aktualisierung) ausgeführt wird.

Anmerkung: Wenn Sie **RECREATE INDEX ON UPDATE** angeben, passiert Folgendes:

- Für die Benutzertabelle werden keine Trigger erstellt.
- Es wird keine Protokolltabelle erstellt.
- Sie können die erweiterte textverwaltete Zwischenspeicherinfrastruktur nicht für den Index konfigurieren.

INDEX DIRECTORY *verzeichnis*

Der Verzeichnispfad, in dem der Textindex gespeichert werden soll. Da das Verzeichnis Indexdaten enthalten wird, müssen Sie sicherstellen, dass der Benutzer-ID des DB2-Instanzeigners die Zugriffsrechte zum Lesen, Schreiben und Ausführen für das Verzeichnis erteilt werden.

Der Standardwert wird der Sicht DB2EXT.DBDEFAULTS mit dem Argument **DEFAULTNAME='INDEXDIRECTORY'** entnommen. Das Unterverzeichnis **NODE<nr>** wird in dem Verzeichnis erstellt, um Indizes auf logischen Knoten eines Servers zu unterscheiden.

Beachten Sie, dass dieses Verzeichnis in einer Umgebung mit partitionierten Datenbanken in jeder physischen Partition vorhanden sein muss.

WORK DIRECTORY *verzeichnis*

Sie können optional ein separates Arbeitsverzeichnis angeben, das zur Speicherung temporärer Dateien bei Indexsuch- und Verwaltungsoperationen verwendet wird. Das Verzeichnis muss vorhanden sein, und der Benutzer-ID des DB2-Instanzeigners müssen die Zugriffsrechte zum Lesen, Schreiben und Ausführen zugeordnet sein.

Der Standardwert wird der Sicht DB2EXT.DBDEFAULTS mit dem Argument **DEFAULTNAME='WORKDIRECTORY'** entnommen. Das Unterverzeichnis **NODE<nr>** wird in dem Verzeichnis erstellt, um Indizes auf logischen Knoten eines Servers zu unterscheiden.

Beachten Sie, dass dieses Verzeichnis in einer Umgebung mit partitionierten Datenbanken in jeder physischen Partition vorhanden sein muss.

Wenn kein Arbeitsverzeichnis (**WORK DIRECTORY**) angegeben ist, wird ein Verzeichnis mit dem Namen **work** unter **INDEX DIRECTORY** erstellt.

Wenn ein **WORK DIRECTORY** (Arbeitsverzeichnis) angegeben ist, empfiehlt es sich sehr, dass es sich im selben physischen Dateisystem befindet wie das Indexverzeichnis. Wenn diese Empfehlung nicht beachtet wird, verursacht dies massive Leistungseinbußen bei der Indexaktualisierung, da Dateien nicht einfach umbenannt werden können, sondern physisch aus dem Arbeitsverzeichnis in das Indexverzeichnis kopiert werden müssen.

Anmerkung: In **INDEX DIRECTORY** und **WORK DIRECTORY** erstellte Dateien folgen der für den Instanzeigner definierten umask-Einschränkung. Stellen Sie sicher, dass diese umask-Einschränkungen Gruppenlese- und -schreibzugriff für den abgeschirmten Benutzer zulassen.

ADMINISTRATION TABLES IN *tabellenbereichsname*

Der Name des regulären Tabellenbereichs für Verwaltungstabellen, die für den Index erstellt werden. Der Tabellenbereich muss vorhanden sein. Wenn er nicht angegeben wird, wird der Tabellenbereich der Benutzertabelle gewählt, wenn der Index für eine Basistabelle erstellt wird.

Im Fall eines Kurznamens oder einer Sicht wird ein Standardtabellenbereich von DB2 ausgewählt.

Wenn Textindizes auf der Grundlage von Sichten, Kurznamen oder Textindizes für die Suche mit gespeicherten Prozeduren in einer Umgebung mit partitionierten Datenbanken erstellt werden, muss der Tabellenbereich für Verwaltungstabellen auf einem einzigen Knoten angegeben sein und auf diesem Knoten explizit aufgerufen werden.

Um sicherzustellen, dass Sie eine Verbindung zum richtigen Knoten herstellen, verwenden Sie die Umgebungsvariable **DB2NODE**. Beachten Sie, dass die Klausel **ADMINISTRATION TABLES IN** für die Erstellung von Indizes für eine bereichspartitionierte Tabelle verbindlich ist. Andernfalls gibt der Befehl **CREATE INDEX** einen Fehler zurück. Weitere Informationen zu der Fehlermeldung finden Sie im Abschnitt CTE0150E.

CACHE TABLE (*sql-spaltenausdruckliste*)

Eine im Cache gespeicherte Tabelle, die aus den angegebenen Spaltenausdrücken besteht, wird zusätzlich zum Index erstellt. Dieser Cache dient zur Rückgabe der Ergebnismenge einer Suche mit gespeicherten Prozeduren, ohne die Ergebnisse der Volltextsuche mit einer DB2-Tabelle zu verknüpfen. Beachten Sie, dass eine reguläre DB2-Suche unter Verwendung des Volltextindex mit der Funktion **CONTAINS** immer möglich ist.

Definieren Sie die SQL-Spaltenausdrücke mit Spaltennamen ohne Qualifikationsmerkmal der Tabelle, für die der Index erstellt wird. Die zulässigen SQL-Spaltenausdruckstypen sind integrierte und benutzerdefinierte einzigartige Datentypen. Die Spaltennamen in der Ergebnismenge werden durch folgende Regeln bestimmt:

- Wenn sie durch die SQL-Klausel **AS** im Spaltenausdruck explizit genannt werden, wird der angegebene Name verwendet. Beispiel:
CACHE TABLE (C1+C2 AS meinname)
- Wenn eine Spalte der angegebenen Tabelle ohne die Klausel **AS** verwendet wird, wird der Name der Spalte verwendet. Beispiel:
CACHE TABLE(C1)
- Wenn ein Ausdruck ohne **AS** verwendet wird und nicht auf eine angegebene Spalte verweist, meldet der Befehl **CREATE INDEX** einen Fehler.
- Spaltennamen müssen eindeutig sein.

Der Datentyp **CLOB** wird nicht als Cachedatentyp unterstützt. Diesen Datentyp müssen Sie in den Datentyp **VARCHARS** umsetzen.

Anmerkung: Beachten Sie, dass der Befehl **CREATE INDEX** einen Fehler liefert, wenn die Spaltennamen in der Ergebnismenge nicht disjunkt sind. Beachten Sie außerdem, dass die im Cache gespeicherte Tabelle nach der Erstellung nicht implizit aktiviert wird. Zum Beispiel ist eine Suche mit gespeicherten Prozeduren nicht möglich, bevor der Befehl **DB2TEXT ACTIVATE CACHE** ausgeführt wird.

Diese Option kann in einer Umgebung mit partitionierten Datenbanken nur verwendet werden, wenn die Benutzertabelle in einem Tabellenbereich gespeichert wird, der sich in einer einzigen Partition befindet. Sie kann nicht verwendet werden, wenn die Standardkonfiguration für die textverwaltete Zwischenspeicherinfrastruktur für die Tabelle auf den Wert 'ON' (Ein) gesetzt ist und die Konfiguration für den Textindex nicht manuell inaktiviert wurde.

PERSISTENT IN *verzeichnis*

Gibt an, dass der Cache auch persistent erstellt wird. Ein persistenter Cache kann nach einer Inaktivierung oder einem Systemwiederanlauf

schneller als ein nicht persistenter Cache reaktiviert werden. Der persistente Cache wird im angegebenen Verzeichnis gespeichert.

Beachten Sie, dass bei keiner Angabe des Verzeichnisses der Standardwert der Sicht `db2ext.dbdefaults` mit dem Argument `DEFAULTNAME='CACHEDIRECTORY'` entnommen wird.

TEMPORARY

Gibt an, dass der Cache nicht persistent gespeichert wird. Wenn weder `PERSISTENT` noch `TEMPORARY` angegeben wird, wird der Standardwert der Sicht `DB2EXT.DBDEFAULTS` mit dem Argument `DEFAULTNAME='USEPERSISTENTCACHE'` entnommen.

MAXIMUM CACHE SIZE *speichergröße*

Gibt die maximale Größe der im Cache gespeicherten Tabelle an, die bei der Ausführung des Befehls **DB2TEXT ACTIVATE CACHE** zu erstellen ist. Der Parameter *speichergröße* ist in MB als positive ganze Zahl (INTEGER) anzugeben. Für *speichergröße* ist kein Standardwert vorhanden. Wenn der Wert zu klein ist, schlägt der Befehl `ACTIVATE CACHE` fehl. Die tatsächliche Cachegröße wird bei der Ausführung des Befehls **ACTIVATE CACHE** berechnet.

Die maximale Cachegröße für die verschiedenen 32-Bit-Plattformen:

- Windows: 1024 MB (1 GB = 1073741824 Bytes)
- Linux: 2048 MB (2 GB = 2147483647 Bytes)

Bei 64-Bit-Installationen ist die maximale Größenbegrenzung für den Cache vom verfügbaren Speicher abhängig.

PCTFREE *prozentsatz*

Gibt den Prozentsatz des Caches an, der für weitere Dokumente freizuhalten ist. Der Prozentsatz muss eine ganze Zahl kleiner als 100 und größer oder gleich 0 sein. Wenn er nicht angegeben wird, wird der Standardwert der Sicht `db2ext.dbdefaults` mit dem Argument `DEFAULTNAME='PCTFREE'` entnommen. Der Standardwert beträgt 50%.

INITIAL SEARCH RESULT ORDER(*sql_sortieren_nach_liste*)

Gibt die Reihenfolge an, die bei der Anfangsindexierung zum Abrufen des Inhalts der Benutzertabelle verwendet wird. Wenn Sie diese Option verwenden und die dynamische Rangordnung von Volltextsuchergebnissen überspringen, werden Dokumente in ihrer Indexierungsreihenfolge wie in der im Cache gespeicherten Ergebnistabelle zurückgegeben.

Sie können mit vorsortierten Indizes und vordefinierter Suchergebnisreihenfolge nur arbeiten, wenn Sie die Suchschnittstelle für gespeicherte Prozeduren verwenden. Beispiel: `INITIAL RESULT ORDER(length(column1) asc, column2+column3 desc)`.

Die Funktion für eine vordefinierte Suchergebnisreihenfolge ist für die skalaren SQL-Suchfunktionen und die Tabellenwertfunktion nicht implementiert.

Anmerkung: Die Indexreihenfolge kann für neue oder geänderte Dokumente nach einer Teilaktualisierung **nicht** gewährleistet werden.

KEY COLUMNS FOR INDEX ON VIEW (*sql-spaltennamensliste*)

Wenn Indizes für Sichten erstellt werden, muss die Klausel **KEY COLUMNS FOR INDEX ON VIEW** angegeben werden. Ansonsten darf diese Klausel NICHT angegeben werden. Die Liste der Spaltennamen gibt die Spalten an, die eine Zeile in der Sicht EINDEUTIG identifizieren.

Da diese Eindeutigkeit nicht wie bei Primärschlüsseln durch DB2 überprüft werden kann, ist es die Aufgabe des Benutzers, eine gleichwertige Eindeutigkeit sicherzustellen. Die angegebenen Spalten werden verwendet, um die Spalte PK01 der Protokolltabelle für den Index zu füllen.

INDEX CONFIGURATION (*optionswert*), ...

Dies sind die Konfigurationswerte für den Index. Die Standardwerte sind unterstrichen.

Option	Werte	Beschreibung
TreatNumbersAsWords	<u>0</u> oder 1	Interpretiert auch dann Folgen von Ziffern als separate Wörter (wenn auf 1 gesetzt), wenn sie neben Buchstaben stehen. Der Standardwert 0 bedeutet, dass tea42at5 als ein einziges Wort betrachtet wird.
IndexStopWords	0 oder <u>1</u>	Ignoriert oder beachtet Stoppwörter beim Indexieren. Durch den Standardwert 1 werden alle Textelemente einschließlich der Stoppwörter indiziert. Gegenwärtig ist die Stoppwörterliste eine UCS-2-Datei mit dem Namen <sprache>.tsw im Verzeichnis <instanz>/sql11ib/db2ext/resources. Änderungen an dieser Datei sind nach der Erstellung des Index wirkungslos . Beachten Sie außerdem, dass <sprache> der Wert für LANGUAGE aus dem Befehl CREATE INDEX ist.
UpdateDelay	Sekunden	Gibt die Zeitdauer in Sekunden an, die für Teilaktualisierungen ohne Erfassungstabellen benötigt wird. Nur Einträge, die älter als diese Zeitdauer sind, werden aus der Protokolltabelle entnommen. Dies dient zur Vermeidung verlorener Aktualisierungen. Ein Beispiel wären Dokumentänderungen, die in Transaktionsszenarios, in denen Benutzertransaktionen zu Problemen mit Aktualisierungsbefehlen führen, nicht im Index wiedergegeben werden. Aus diesem Grund sollte für den Parameter UpdateDelay die maximal zulässige Zeitdauer einer Benutzerschreibtransaktion für die Tabelle angegeben werden, für die der Index erstellt wurde.
IgnoreEmptyDocs	<u>0</u> oder 1	Wenn für IgnoreEmptyDocs der Wert 1 definiert wird, werden leere Dokumente (mit einer Inhaltsgröße von 0 oder einem Nullwert) im Index nicht dargestellt., Wenn diese Option verwendet wird und der Dokumentinhalt null (leer) ist, werden bei der nächsten Teilaktualisierung die Dokumente aus dem Index gelöscht.
AuxLog	ON oder OFF	Steuert die Erstellung der zusätzlichen Protokollinfrastruktur, um Änderungen zu erfassen, die von einem Trigger nicht erkannt werden. Die Standardeinstellung für bereichspartitionierte Tabellen ist ON (Ein), andernfalls ist die Einstellung OFF (Aus). Der Standardwert kann in der Standardtabelle geändert werden, indem 'AuxLogNorm' für nicht-bereichspartitionierte Tabellen bzw. 'AuxLogPart' für bereichspartitionierte Tabellen festgelegt wird.

CONNECT TO *datenbankname*

Der Name der Datenbank, die das Ziel für diesen Befehl darstellt. Dieser Parameter kann ausgelassen werden, wenn die Umgebungsvariable

DB2DBDFT definiert ist und der Benutzer den Befehl auf dem Server ausführt. Beachten Sie, dass die Benutzer-ID über die erforderlichen DB2-Berechtigungen verfügen muss.

USER *benutzer-id* **USING** *kennwort*

Verwenden Sie eine *benutzer-id* und ein *kennwort* zur Herstellung einer Verbindung zur Datenbank. Wenn Sie diese Angaben nicht machen, wird versucht, mit der aktuellen Benutzer-ID ohne Kennwort eine Verbindung herzustellen.

Änderungen an der Datenbank

- Die Net Search Extender-Katalogsichten werden geändert.
- Eine Indexprotokolltabelle wird im angegebenen Tabellenbereich erstellt. Dies geschieht nur, wenn die Option **RECREATE INDEX** und die Erfassungstabelle nicht angegeben werden.
- Eine Indexereignistabelle wird im angegebenen Tabellenbereich erstellt.
- Die Erstellung der Trigger für die Benutzertexttabelle wird bis zur ersten Aktualisierung verzögert (nur wenn die Option **RECREATE INDEX** nicht angegeben und keine Erfassungstabelle verwendet wird).
- Wenn eine Replikationserfassungstabelle verwendet wird, werden die Erfassungssteuertabellen wie folgt geändert:
 - Für die Tabellen IBMSNAP_PRUNCTNL und IBMSNAP_PRUNE_SET wird eine INSERT-Operation ausgeführt.

Die Einträge in diesen Tabellen werden durch die Spalten eindeutig identifiziert:

- APPLY_QUAL='NSE' || <DB2-Instanz_auf_der_NSE_ausgefuehrt_wird>
- SET_NAME= <interne_index-ID>
- TARGET_SERVER=<name_der_DB2-datenbank_die_das_ziel_der_DB2TEXT-operation_ist>
- Wenn die textverwaltete Zwischenspeicherinfrastruktur für den Index konfiguriert ist, werden eine Ankertabelle und eine Zwischenspeichertabelle erstellt, um Änderungsinformationen für Dokumente zu erfassen.

Änderungen am gemeinsamen Speicher

Verzögert bis zur Ausführung des ACTIVATE-Befehls: Wenn die Klausel **CACHE TABLE** verwendet wird, wird ein Cache für die Ergebnistabelle im *gemeinsamen Speicher* erstellt.

Änderungen am Dateisystem

- Im Index-, Arbeits- und Cacheverzeichnis werden Unterverzeichnisse namens **NODE<nr>** erstellt.
- Das Verzeichnis *interner_indexname* wird im Verzeichnis *indexverzeichnis/NODE<nr>* erstellt, wobei sich *indexverzeichnis* auf den Wert des entsprechenden Parameters dieses Befehls und **NODE<nr>** auf die Partitionsnummer in einer Umgebung mit partitionierten Datenbanken bezieht.

Hinweise zur Verwendung

Zur Erstellung eines Volltextindex ist ein Primärschlüssel in der Benutzertabelle erforderlich. Ab DB2 Net Search Extender Version 9.1 kann ein mehrspaltiger DB2-Primärschlüssel ohne Typeinschränkungen verwendet werden. Bei der Tabellenwertsuche sind jedoch keine zusammengesetzten Primärschlüssel zulässig.

Die Anzahl der Primärschlüsselspalten ist auf 62, die Gesamtlänge aller Primärschlüsselspalten auf 1007 Byte für Tabellenbereiche mit einer Seitengröße von 4 K, auf 2031 Byte für Tabellenbereiche mit einer Seitengröße von 8 K, auf 4079 Byte für Tabellenbereiche mit einer Seitengröße von 16 K und auf 4096 Byte für Tabellenbereiche mit einer Seitengröße von 32 K begrenzt. Wenn der Primärschlüssel aus mehreren Spalten besteht, müssen die angegebenen Begrenzungen für jede zusätzliche Spalte um 2 Byte verringert werden.

- Die Gesamtgröße der SQL-Ausdrücke für ATTRIBUTES, CACHE TABLE und INITIAL SEARCH RESULT ORDER darf 24 KB nicht überschreiten.
- Anfangsindexaktualisierung werden immer in einer logischen Transaktion durchgeführt. In diesem Fall gibt es keinen Commitzähler.

Auf Windows-Plattformen muss sich **modelfilepath** auf dem lokalen Dateisystem befinden. Über Netze zugeordnete Laufwerke unter Windows werden nicht unterstützt.

Anmerkung:

Wenn ein Primärschlüssel aus mehreren Spalten besteht, müssen Sie sicherstellen, dass die in der Primärschlüsseldefinition angegebene Spaltenreihenfolge mit der Reihenfolge in der Anweisung **CREATE TABLE** genau übereinstimmt.

Beachten Sie, dass nach der Erstellung des Index die Länge der Primärschlüsselspalten oder der Sichtschlüsselspalten nicht durch **ALTER TABLE**-Befehle geändert werden darf.

Die Synchronisation zwischen der Benutzertabelle, dem Volltextindex und der im Cache gespeicherten Ergebnistabelle wird während der Ausführung des Befehls **UPDATE INDEX** durchgeführt.

Befehl DEACTIVATE CACHE

Gibt eine im Cache gespeicherte Tabelle wieder frei. Ein persistenter Cache bleibt erhalten, sodass er im nächsten **ACTIVATE**-Befehl wiederverwendet werden kann. Bis zur nächsten Aktivierung sind Suchoperationen über eine gespeicherte Prozedur mit dem inaktivierten Cache nicht möglich.

Berechtigung

Die Berechtigungs-ID der Anweisung muss mindestens eine der folgenden Berechtigungen umfassen:

- Das Zugriffsrecht CONTROL für die Tabelle oder den Kurznamen, für die bzw. den der Textindex definiert wurde.
- Die Berechtigung DBADM.

Voraussetzung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS verfügen. In Version 9.7 verfügt der SYSADM nicht mehr über die Berechtigung SECADM oder DBADM. Der SECADM muss dem Instanzeigner die Berechtigung DBADM und das Zugriffsrecht DATAACCESS explizit erteilen, bevor der Befehl **DEACTIVATE CACHE** ausgeführt werden kann.

Befehlssyntax

```
▶▶ DEACTIVATE CACHE FOR INDEX indexschema-".." indexname FOR TEXT ▶▶
```

└─|verbindungsoptionen|─┘

verbindungsoptionen:

```
└─|CONNECT TO datenbankname USER benutzer-id USING kennwort|─┘
```

Befehlsparameter

indexschema

Das Schema des Textindex, wie es im Befehl **CREATE INDEX** angegeben wurde. Wenn kein Schema angegeben wird, wird die Benutzer-ID der DB2-Verbindung als Schemaname verwendet.

indexname

Der Name des Textindex, wie er im Befehl **CREATE INDEX** angegeben wurde.

CONNECT TO *datenbankname*

Der Name der Datenbank, die das Ziel für diesen Befehl darstellt. Dieser Parameter kann ausgelassen werden, wenn die Umgebungsvariable **DB2DBDFT** definiert ist und der Benutzer den Befehl auf dem Server ausführt. Beachten Sie, dass die Benutzer-ID über die erforderlichen DB2-Berechtigungen verfügen muss.

USER *benutzer-id* **USING** *kennwort*

Verwenden Sie eine *benutzer-id* und ein *kennwort* zur Herstellung einer Verbindung zur Datenbank. Wenn Sie diese Angaben nicht machen, wird versucht, mit der aktuellen Benutzer-ID ohne Kennwort eine Verbindung herzustellen.

Hinweise zur Verwendung

Beachten Sie, dass dieser Befehl nicht abgesetzt werden kann, wenn einer der folgenden Befehle momentan für den Index ausgeführt wird:

- **ACTIVATE CACHE**
- **DEACTIVATE CACHE**
- **UPDATE INDEX**
- **ALTER INDEX**
- **DROP INDEX**
- **CLEAR EVENTS**

Anmerkung: Nach der Inaktivierung eines persistenten Caches ist der Zugriff auf den Cache für eine Suche mit gespeicherten Prozeduren nicht möglich. Allerdings kann dieser Befehl verwendet werden, wenn eine schnelle Aktivierung (**ACTIVATE**) ausgeführt werden soll, sofern in der Zwischenzeit keine Aktualisierung erfolgt ist.

In diesem Fall wird der persistente Cache mit dem Befehl **ACTIVATE CACHE** automatisch völlig neu erstellt.

Befehl DROP INDEX

Löscht einen Volltextindex für eine Textspalte. Wenn der Cache für den Index aktiviert ist, wird er mit diesem Befehl gelöscht.

Berechtigung

Die Berechtigungs-ID der Anweisung muss mindestens eine der folgenden Berechtigungen umfassen:

- Das Zugriffsrecht CONTROL für die Tabelle oder den Kurznamen, für die bzw. den der Textindex definiert wurde.
- Die Berechtigung DBADM.

Voraussetzung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS verfügen. In Version 9.7 verfügt der SYSADM nicht mehr über die Berechtigung SECADM oder DBADM. Der SECADM muss dem Instanzeigner die Berechtigung DBADM und das Zugriffsrecht DATAACCESS explizit erteilen, bevor der Befehl **DROP INDEX** ausgeführt werden kann.

Befehlssyntax

```
►► DROP INDEX [indexschema-"."] indexname FOR TEXT ►►  
|verbindungsoptionen|
```

verbindungsoptionen:

```
|CONNECT-TO datenbankname |  
|USER benutzer-id USING kennwort |
```

Befehlsparameter

indexschema

Das Schema des Textindex, wie es im Befehl **CREATE INDEX** angegeben wurde. Wenn kein Schema angegeben wird, wird die Benutzer-ID der DB2-Verbindung als Schemaname verwendet.

indexname

Der Name des Index, wie er im Befehl **CREATE INDEX** angegeben wurde. Zusammen mit dem Schemanamen dient dieser Name zur eindeutigen Identifikation des Volltextindex in einer Datenbank.

CONNECT TO *datenbankname*

Der Name der Datenbank, die das Ziel für diesen Befehl darstellt. Dieser Parameter kann ausgelassen werden, wenn die Umgebungsvariable

DB2DBDFT definiert ist und der Benutzer den Befehl auf dem Server ausführt. Beachten Sie, dass die Benutzer-ID über die erforderlichen DB2-Berechtigungen verfügen muss.

USER *benutzer-id* **USING** *kennwort*

Verwenden Sie eine *benutzer-id* und ein *kennwort* zur Herstellung einer Verbindung zur Datenbank. Wenn Sie diese Angaben nicht machen, wird versucht, mit der aktuellen Benutzer-ID ohne Kennwort eine Verbindung herzustellen.

Hinweise zur Verwendung

Der Index wird gelöscht, und zwar unabhängig vom Aktivierungsstatus der zugehörigen im Cache gespeicherten Tabelle.

Beachten Sie, dass der Befehl nicht abgesetzt werden darf, wenn einer der folgenden Befehle momentan für den Index ausgeführt wird:

- **UPDATE INDEX**
- **CLEAR EVENTS**
- **ALTER INDEX**
- **ACTIVATE CACHE**
- **DEACTIVATE CACHE**
- **DROP INDEX**

Anmerkung: Indizes müssen manuell gelöscht werden, bevor oder nachdem die Benutzertabelle in DB2 gelöscht wird. Anderenfalls werden die Indexverzeichnisse nicht korrekt bereinigt.

Änderungen an der Datenbank

- Die Net Search Extender-Katalogsichten werden geändert.
- Der DB2-Index wird gelöscht.
- Das Indexprotokoll, die Zwischenspeicher- und die Ereignistabellen werden gelöscht.
- Trigger für die Benutzertexttabelle werden gelöscht.

Wenn die Replikationserfassungstabellen verwendet werden, werden die Einträge in den Tabellen `IBMSNAP_PRUNE_SET` und `IBMSNAP_PRUNCTRNL` entfernt.

Änderungen am gemeinsamen Speicher

Die im Cache gespeicherte Tabelle wird gelöscht.

Änderungen am Dateisystem

- Das Verzeichnis *interner_indexname* wird im Index- und im Arbeitsverzeichnis des gelöschten Index gelöscht.
- Ein persistenter Cache für den Index wird gelöscht.

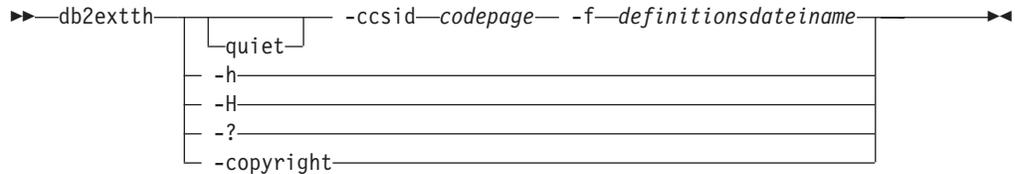
Befehl DB2EXTTH

Dieses unabhängige Dienstprogramm kompiliert eine Thesaurusdefinitionsdatei. Nach der Ausführung des Thesauruscompilers können die THESAURUS-Optionen der Suchargumentsyntax verwendet werden.

Berechtigung

Keine. Dieser Befehl ist nicht notwendigerweise auf den Tabelleneigner beschränkt, ist jedoch nur im Kontext von Abfragen sinnvoll.

Befehlssyntax



Befehlsparameter

`-f definitionsdateiname`

Der Name der Datei, in der die Thesaurusdefinition enthalten ist. Der Dateiname muss entweder den absoluten oder den relativen Pfad zu der Datei enthalten. Der Dateiname ist auf 8+3 Zeichen begrenzt, wobei die Angabe der Erweiterung optional ist.

Das Thesauruswörterverzeichnis wird im gleichen Verzeichnis wie die Definitionsdatei und unter dem gleichen Namen erstellt. Der einzige Unterschied besteht darin, dass das Wörterverzeichnis die folgenden Erweiterungen hat: `wdf`, `wdv`, `grf`, `grv`, `MEY`, `ROS`, `NEY`, `SOS` und `Ikn`. Dabei steht *n* für eine Ziffer. Beachten Sie, dass vorhandene Thesaurusdateien mit dem gleichen Namen überschrieben werden.

`-ccsid codepage`

Die Codepage, in der die Thesaurusdefinitionsdatei geschrieben ist.

`-quiet` Es werden keine Ausgabeinformationen angezeigt.

`-copyright`

Gibt die interne Buildnummer des Produkts zurück. Geben Sie diese Nummer bei der Meldung von Problemen an.

`-h | -H | -?`

Zeigt Hilfeinformationen an.

Hinweise zur Verwendung

Verwenden Sie diesen Befehl, um eine Thesaurusdefinitionsdatei in ein binäres Thesaurusdefinitionsformat umzuwandeln.

Befehl RESET PENDING

Wenn Sie die erweiterte textverwaltete Zwischenspeicherinfrastruktur verwenden, wird die Zwischenspeichertabelle durch bestimmte Befehle in den Modus 'Anstehend' versetzt, wodurch andere Datenbank- oder Textsuchoperationen blockiert werden. Sie können den Befehl **RESET PENDING** verwenden, um SET INTEGRITY für alle textverwalteten Zwischenspeichertabellen auszuführen, die einer bestimmten Tabelle zugeordnet sind. Sie müssen nicht alle Textindizes und zugeordneten Zwischenspeichertabellen suchen, um den Befehl **SET INTEGRITY** für alle Tabellen auszuführen.

Nach dem Aufheben der Zuordnung zu einer Datenpartition müssen Sie den Befehl **RESET PENDING** ausführen, um den Inhalt der Zwischenspeichertabelle zu aktualisieren.

Berechtigung

Sie müssen für die Tabelle das Zugriffsrecht **CONTROL** besitzen.

Wenn der SECADM die Zugriffsberechtigungen für eine oder mehrere textverwaltete Zwischenspeichertabellen ändert, die einer Basistabelle zugeordnet sind, benötigen Sie immer noch das Zugriffsrecht, mit dem Sie die Anweisung **SET INTEGRITY** für eine oder mehrere textverwaltete Zwischenspeichertabellen ausführen können.

Erforderliche Verbindung

Dieser Befehl muss vom DB2-Datenbankserver aus ausgegeben werden.

Befehlssyntax

```
▶▶—RESET PENDING FOR TABLE table-schema.table-name FOR TEXT—————▶▶  
|_____|  
|_|verbindungsoptionen|_|
```

Verbindungsoptionen:

```
|_____|  
|_|CONNECT TO—datenbankname_____|  
|_|USER—benutzer-id—USING—kennwort|_|
```

Befehlsparameter

tabellenname

Der Name der Tabelle, für die die textverwaltete Zwischenspeicherinfrastruktur hinzugefügt wurde und die eine Integritätsverarbeitung erfordert.

Tabellenschema

Das Schema der Tabelle, für die ein Befehl ausgeführt wurde, der zum Modus 'Anstehend' für abhängige Tabellen führt.

Hinweise zur Verwendung

Verwenden Sie den Befehl **RESET PENDING**, nachdem Sie einen Befehl ausgeführt haben, mit dem abhängige Tabellen in den Modus 'Anstehend' versetzt werden, wie z. B. den Befehl **LOAD** mit dem Parameter **INSERT**, oder einen Befehl, der die Anweisung **SET INTEGRITY** erfordert, um abhängige Tabellen zu aktualisieren, wie z. B. **ALTER TABLE...DETACH**.

Befehl UPDATE INDEX

Prüft den für die Aktualisierungsoperation benötigten Plattenspeicherplatz, bevor die eigentliche Indexaktualisierung gestartet wird. Ist die Prüfung erfolgreich, wird der Indexaktualisierungsprozess fortgesetzt.

Zweck

Der erforderliche Plattenspeicherplatz wird für einen einzigen Aktualisierungsprozess berechnet. Eine Indexaktualisierung wird beendet, falls der geschätzte Speicherplatz nicht verfügbar ist. Weitere Informationen finden Sie in Teil 7, „Planungsaspekte“, auf Seite 57.

Der Indexierungsprozess wird gestartet, indem der Index auf den neuesten Stand gebracht wird, sodass er den aktuellen Inhalt der Textspalten widerspiegelt, denen der Index zugeordnet ist.

Während der Ausführung der Aktualisierung ist das Suchen mit dem Vergleichselement CONTAINS möglich. Bei einem Index mit einer aktivierten, im Cache gespeicherten Ergebnistabelle ist das Suchen mit gespeicherten Prozeduren während der Aktualisierung ebenfalls möglich. Allerdings können die Spalten in der im Cache gespeicherten Tabelle neue Werte aufweisen, obwohl der geänderte Text im Volltextindex noch nicht festgeschrieben ist (Commit).

Bei Verwendung der Option **RECREATE INDEX ON UPDATE** im Befehl **CREATE INDEX** wird der Inhalt des Index vor der Neuerstellung gelöscht. Bis zum Abschluss der Aktualisierung werden leere Ergebnisse zurückgegeben.

Berechtigung

Die Berechtigungs-ID der Anweisung muss mindestens eine der folgenden Berechtigungen umfassen:

- Das Zugriffsrecht CONTROL für die Tabelle oder den Kurznamen, für die bzw. den der Textindex definiert wurde.
- Das Zugriffsrecht DATAACCESS.

Voraussetzung

Der Instanzeigner muss über die Berechtigung DBADM und das Zugriffsrecht DATAACCESS verfügen. In Version 9.7 verfügt der SYSADM nicht mehr über die Berechtigung SECADM oder DBADM. Der SECADM muss dem Instanzeigner die Berechtigung DBADM und das Zugriffsrecht DATAACCESS explizit erteilen, bevor der Befehl **UPDATE INDEX** ausgeführt werden kann.

Befehlssyntax

```
►► UPDATE INDEX [indexschema-". " ] indexname FOR TEXT [REORGANIZE ]
[COMMITCOUNT—anzahl ] [USING UPDATE MINIMUM ] [SKIPDISKSPACECHECK ]
[verbindungsoptionen ]
```

verbindungsoptionen:

```
[CONNECT TO—datenbankname [USER—benutzer-id—USING—kennwort ] ]
```

Befehlsparameter

indexschema

Das Schema des Textindex. Dies entspricht der Angabe im Befehl **CREATE INDEX**. Wenn kein Schema angegeben wurde, wird die Benutzer-ID der DB2-Verbindung verwendet.

indexname

Der Name des Textindex. Dies entspricht der Angabe im Befehl **CREATE INDEX**.

REORGANIZE

Wenn eine Textspalte häufig aktualisiert wird, können nachfolgende Aktualisierungen am Index ineffizient werden. Um den Aktualisierungsprozess wieder effizient zu machen, kann der Index reorganisiert werden. Anhand der Sicht DB2EXT.TEXTINDEXES können Sie feststellen, ob ein Index reorganisiert werden muss.

Verwenden Sie die Option **REORGANIZE AUTOMATIC** des Befehls **CREATE INDEX**, wenn Sie eine manuelle Überprüfung und Reorganisation des Index vermeiden wollen.

Anmerkung: Der Reorganisationsprozess findet nach einer regulären Aktualisierung statt.

USING UPDATE MINIMUM

Diese Option verwendet die Einstellungen für **UPDATE MINIMUM** aus dem Befehl **CREATE INDEX** und startet eine inkrementelle Aktualisierung (Teilaktualisierung), sofern die angegebene Anzahl von Änderungen erreicht wurde. Wenn die textverwaltete Zwischenspeicherinfrastruktur konfiguriert ist, wird die Anzahl der Änderungen für diese Bestimmung kombiniert. Die Standardfunktionsweise besteht darin, die Aktualisierung bedingungslos zu starten.

Bei partitionierten Datenbanken wird **UPDATE MINIMUM** in jeder Partition geprüft.

COMMITCOUNT *anzahl*

Ein ganzzahliger Wert (INTEGER) ≥ 0 legt die Anzahl von Dokumenten fest, die in einer Transaktion durch die Suchsteuerkomponente sowie durch DB2 bei Indexteilaktualisierungen verarbeitet werden. Dieser Wert kann mit dem Befehl **ALTER INDEX** geändert werden.

Allerdings gibt es bei Anfangsaktualisierungen, wie der ersten Aktualisierung nach Ausführung des Befehls **CREATE INDEX** oder einer beliebigen Aktualisierung mit der Option **RECREATE INDEX ON UPDATE**, nur eine einzige logische Transaktion, bei der der Wert für **COMMITCOUNT** ignoriert wird.

Die Verwendung eines **COMMITCOUNT** (Commitzählers) ungleich null ist nicht empfehlenswert, da dies massive negative Auswirkungen auf den Indexdurchsatz hat.

SKIPDISKSPACECHECK

Wenn diese Option angegeben wird, wird die Plattenspeicherplatzüberprüfung für eine inkrementelle Indexaktualisierung übersprungen. Standardmäßig wird der Plattenspeicherplatz bei jeder Indexaktualisierungsoperation überprüft.

CONNECT TO *datenbankname*

Der Name der Datenbank, die das Ziel für diesen Befehl darstellt. Dieser Parameter kann ausgelassen werden, wenn die Umgebungsvariable

DB2DBDFT definiert ist und der Benutzer den Befehl auf dem Server ausführt. Beachten Sie, dass die Benutzer-ID über die erforderlichen DB2-Berechtigungen verfügen muss.

USER *benutzer-id* **USING** *kennwort*

Verwenden Sie eine *benutzer-id* und ein *kennwort* zur Herstellung einer Verbindung zur Datenbank. Wenn Sie diese Angaben nicht machen, wird versucht, mit der aktuellen Benutzer-ID ohne Kennwort eine Verbindung herzustellen.

Hinweise zur Verwendung

Dieser Befehl wird synchron ausgeführt. Er startet die Aktualisierungsverarbeitung auf allen erforderlichen logischen/physischen DB2-Partitionen in einer Umgebung mit partitionierten Datenbanken. Die Verarbeitungsdauer ist von der Anzahl zu indexierender Dokumente sowie von der Anzahl bereits indexierter Dokumente abhängig. Der Status der Aktualisierung kann anhand einer Sicht, die für jeden Index erstellt wird, abgefragt werden. Der Name dieser Sicht ist der Spalte **EVENT-VIEWNAME** der Sicht **DB2EXT.TEXTINDEXES** zu entnehmen.

Es gibt zwei verschiedene Optionen, um die Anzahl der festgeschriebenen Dokumente anzuzeigen, die verarbeitet wurden. Verwenden Sie die Sicht **DB2EXT.TEXTINDEXES (NUMBER_DOCS)**, um festzustellen, wie viele Dokumente im Index festgeschrieben wurden. Verwenden Sie die Ereignissicht, die dem Index zugeordnet ist, für Informationen zum Starten, Commit von Änderungen und Beenden der Aktualisierungsverarbeitung.

Um die Anzahl der Dokumente anzuzeigen, die verarbeitet wurden, während eine Indexaktualisierung noch aktiv ist, verwenden Sie den Befehl **CONTROL LIST ALL LOCKS FOR INDEX**.

Anmerkung: In den Sichten werden nur Informationen der verbundenen Partition angezeigt.

Bei Teilaktualisierungen in einer Basistabelle mit mehreren physischen Partitionen muss die Uhrzeit in allen Partitionen synchronisiert werden. Andernfalls gehen Aktualisierungen möglicherweise verloren oder werden gar nicht ausgeführt.

Sie können den Befehl nicht absetzen, wenn einer der folgenden Befehle momentan für den Index ausgeführt wird:

- **CLEAR EVENTS**
- **ALTER INDEX**
- **DROP INDEX**
- **ACTIVATE CACHE**
- **DEACTIVATE CACHE**
- **UPDATE INDEX**

Nach der Aktualisierung eines Index mit einer inaktivierten im Cache gespeicherten persistenten Ergebnistabelle wird der persistente Cache gelöscht, sodass der nächste Befehl **ACTIVATE CACHE** ihn auf der Grundlage des Datenbankinhalts erneut erstellt.

Wenn der Benutzer diesen Befehl unterbricht, werden alle an der Aktualisierungsfunktion beteiligten Prozesse gestoppt. Wenn ein **COMMITCOUNT** (Commitzähler) in einer Teilaktualisierung verwendet wurde, wurden einige Aktualisierungen

möglicherweise festgeschrieben und sind im Index sichtbar, während für andere möglicherweise ein neuer Befehl UPDATE erforderlich ist. Wenn der Indexaktualisierungsprozess erzwungen unterbrochen wird, wird der Index möglicherweise beschädigt.

Zum Stoppen der automatischen Aktualisierung eines Index suchen Sie nach dem DB2-Instanzeignerprozess, der den Befehl zur Indexaktualisierung in der Partition ausführt, die für Aktualisierungsservices verwendet wird. Stoppen Sie diesen Prozess und die Aktualisierungsverarbeitung in allen Partitionen.

Anmerkung: Da der Befehl in zwei getrennten Phasen, bei der Indexerstellung in allen Partitionen und bei den Anfangsaktualisierungen des Index, ausgeführt wird, müssen Sie einen **db2text drop index**-Befehl absetzen, um sicherzustellen, dass der Index nicht teilweise verfügbar ist. Wenn dieser Befehl nicht abgesetzt wird, führt die nächste Aktualisierung, die durch einen manuellen Aktualisierungsbefehl oder durch die Option UPDATE FREQUENCY ausgelöst werden kann, eine vollständige Neuindexierung durch, um einen konsistenten Zustand sicherzustellen.

Änderungen an der Datenbank

- Es werden Einfügungen an der Ereignistabelle durchgeführt.
- Löschungen aus der Indexprotokolltabelle und der textverwalteten Zwischenspeichertabelle für den Index, falls konfiguriert.

Bei Verwendung der Replikationserfassungstabellen werden in der Datenbank die folgenden Änderungen durchgeführt.

- Vor der Anfangsaktualisierung wird ein Signal in der Tabelle IBMSNAP_SIGNAL hinzugefügt.
- Der Synchronisationspunkt von IBMSNAP_PRUNE_SET wird nach der Teilaktualisierung geändert.

Befehl HELP

Zeigt die Liste der verfügbaren **DB2TEXT**-Befehle oder die Syntax eines einzelnen **DB2TEXT**-Befehls an.

Berechtigung

Keine.

Befehlssyntax



Befehlsparameter

HELP | ?

Stellt Hilfeinformationen für den angegebenen Befehl oder Ursachencode bereit.

befehl Die ersten Schlüsselwörter, die einen **DB2TEXT**-Befehl identifizieren:

- ENABLE
- DISABLE

- CREATE
- DROP
- ALTER
- UPDATE
- CLEAR
- START
- STOP
- CONTROL
- ACTIVATE
- DEACTIVATE
- RESET PENDING

ursachencode

Der Ursachencode aus dem Net Search Extender-Befehl wie z. B. CTE0192.

Hinweise zur Verwendung

Wenn mehr als das erste Schlüsselwort angegeben wird, wird der Rest ignoriert und die Syntax des erkannten Befehls angezeigt.

Wenn kein Parameter *befehl* nach **?** oder **HELP** (bzw. überhaupt kein Parameter) angegeben wird, listet **DB2TEXT** alle verfügbaren **DB2TEXT**-Befehlsparameter auf.

Befehl COPYRIGHT

Zeigt Produkt- und Copyright-Informationen zu Net Search Extender an.

Berechtigung

Keine.

Befehlssyntax



Befehlsparameter

COPYRIGHT | LEVEL

Zeigt die Copyright-Version, die Versionsnummer und die Buildinformationen für das Produkt an.

Kapitel 58. Befehlsreferenz für die Installation und Deinstallation von Net Search Extender unter UNIX

Für die Installation und Deinstallation von Net Search Extender unter UNIX müssen Sie die entsprechende Syntax für diese Prozesse kennen.

Dieser Abschnitt beschreibt die Syntax des Installations- und des Deinstallationsbefehls für Net Search Extender unter UNIX. Auch der Befehl **db2nse1s** wird erläutert, mit dem die installierten Kopien des Produkts Net Search Extender ab DB2 Version 9.1 angezeigt werden.

Befehl db2nse_install

Mit diesem Befehl wird eine neue Version von Net Search Extender auf einem UNIX-Betriebssystem installiert.

Berechtigung

Rootberechtigung

Befehlssyntax

```
→ db2nse_install paketdatei pfad [-s] [-f] [-v]
```

Befehlsparameter

paketdatei

Der Name der Datei mit dem Produkt Net Search Extender.

pfad

Der DB2-Datenbankpfad, in dem Sie nun Net Search Extender installieren möchten.

- s** Automatische Installation. Es werden Überprüfungen durchgeführt, und eine Protokolldatei wird in das Verzeichnis /tmp geschrieben.
- f** Installation erzwingen. Es findet keine Überprüfung statt.
- v** Zeigt die Programmversion an und beendet das Programm.

Hinweise zur Verwendung

Mit diesem Befehl wird Net Search Extender installiert, das sich in der Programmdatei befand, die als Parameter übergeben wurde. Wenn Sie keine weiteren Parameter angeben, prüft das Installationsprogramm das System auf infrage kommende Kopien der DB2-Datenbanksysteme, in denen Net Search Extender installiert werden kann. Alle DB2-Datenbankinstallationspfade werden aufgeführt; allerdings kann nur eine funktionsfähige Untergruppe ausgewählt werden. Sie werden dazu aufgefordert, einen Installationspfad auszuwählen.

Wenn Sie eine automatische Installation durchführen möchten, ist keine Benutzerinteraktion erforderlich. Das Ergebnis des Installationsprozesses wird als Rückkehrcode an das aufrufende Programm übergeben; dabei kann es sich um eine Befehls-

shell oder ein Shell-Script handeln. Es ist Ihre Aufgabe, den Rückkehrcode ordnungsgemäß zu handhaben. Es wird eine Protokolldatei in das Verzeichnis /tmp geschrieben, in der die Installation dokumentiert wird.

Wenn Sie eine erzwungene Installation durchführen möchten, ist keine Benutzerinteraktion erforderlich; außerdem werden keine zusätzlichen Überprüfungen durchgeführt. Das Ergebnis des Installationsprozesses wird als Rückkehrcode an das aufrufende Programm übergeben; dabei kann es sich um eine Befehlsshell oder ein Shell-Script handeln. Es ist Ihre Aufgabe, den Rückkehrcode ordnungsgemäß zu handhaben. Es wird eine Protokolldatei in das Verzeichnis /tmp geschrieben, in der die Installation dokumentiert wird.

Für die beiden Parameter **-s** und **-f** muss ein Pfad übergeben werden. Für diese Parameter ist kein Standardpfad vorhanden.

Wenn der Parameter **-v** übergeben wird, wird nur die Programmversion angezeigt, und das Programm wird ohne weitere Aktionen beendet.

Befehl db2nse_deinstall

Mit diesem Befehl wird Net Search Extender von UNIX-Betriebssystemen entfernt.

Der Befehl **db2nse_deinstall** befindet sich in jedem Installationsunterverzeichnis einer DB2-Kopie, wo Net Search Extender installiert ist.

Berechtigung

Rootberechtigung

Befehlssyntax

▶▶—db2nse_deinstall -v▶▶

Befehlsparameter

-v Zeigt die Programmversion an, und beendet das Programm.

Hinweise zur Verwendung

Mit diesem Befehl wird Net Search Extender ab DB2 Version 9.1 entfernt. Mit dem Befehl wird die Net Search Extender-Kopie aus dem Verzeichnis entfernt, wo er abgesetzt wird. Andere Net Search Extender-Kopien, die an anderen Positionen installiert sind, werden nicht entfernt.

Befehl db2nsels

Listet alle installierten Kopien von Net Search Extender auf. Dies gilt ab DB2 Version 9.7.

Der Befehl **db2nsels** befindet sich im Verzeichnis /usr/local/bin.

Berechtigung

Rootberechtigung

Befehlssyntax



Befehlsparameter

- c** Zeigt die installierten Net Search Extender-Versionen in einer einfachen kompakten Liste an, wobei Doppelpunkte als Trennungszeichen dienen, und beendet das Programm.
- v** Zeigt die Programmversion an, z. B. 9.7, und beendet das Programm.

Hinweise zur Verwendung

Das Befehlsergebnis wird entweder in einer Tabelle oder einer kompakten Liste angezeigt, wobei die Elemente durch Doppelpunkte getrennt werden. In den Ergebnissen sind der Pfad, die Version sowie die Fixpacknummer der Net Search Extender-Installation enthalten.

Beispiel: Wenn Sie **db2nse1s** ohne Parameter aufrufen, sieht die Ausgabe ähnlich der Folgenden aus:

```
db2nse1s
```

Installationspfad	Version	FP
/opt/ibm/db2/V9.7	9.7.0.0	0
/test/V9.7	9.7.0.0	0

Wenn Sie den Parameter **-c** verwenden, handelt es sich bei der zurückgegebenen Ausgabe um eine kompakte Liste, bei der die einzelnen Informationselemente durch Doppelpunkte getrennt werden. Dieser Ausgabebetyp kann mit einem Programm oder einer Shell-Script einfach gehandhabt werden. Beispiel:

```
db2nse1s -c
```

```
#PATH:VRMF:FIXPACK  
/opt/ibm/db2/V9.7:9.7.0.0:0  
/test/V9.7:9.7.0.0:0
```

Kapitel 59. Syntax für Suchargumente

Ein Suchargument ist die Bedingung, die Sie für die Suche nach Begriffen in Textdokumenten angeben. Das Suchargument umfasst Suchparameter sowie einen oder mehrere Suchbegriffe.

Beispiele für Suchargumente finden Sie in Kapitel 42, „Angeben von SQL-Suchargumenten“, auf Seite 137 und in der Datei `search` im Verzeichnis `samples` von Net Search Extender.

Die folgenden skalaren SQL-Suchfunktionen arbeiten mit Suchargumenten:

CONTAINS

Diese Funktion verwendet ein Suchargument zur Suche nach Text in einem bestimmten Textdokument. Sie liefert den ganzzahligen Wert (INTEGER) 1, wenn das Dokument den gesuchten Text bzw. eine beliebige, im Suchargument angegebene Relation enthält. Anderenfalls liefert sie den Wert 0.

NUMBEROFMATCHES

Diese Funktion verwendet ein Suchargument zum Durchsuchen von Textdokumenten und gibt einen ganzzahligen Wert (INTEGER) zurück, der die Anzahl der gefundenen Übereinstimmungen pro Dokument angibt.

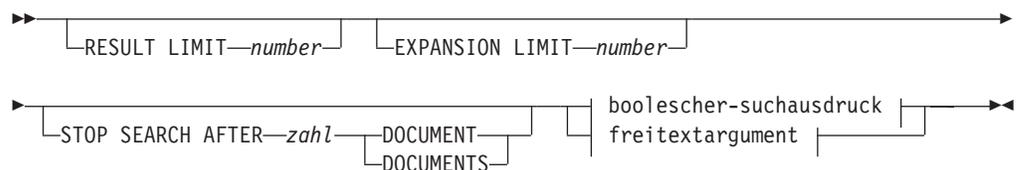
SCORE

Diese Funktion verwendet ein Suchargument zur Suche in Textdokumenten. Sie gibt einen Wert für jedes gefundene Dokument zurück, der angibt, wie gut das gefundene Dokument durch das Suchargument im Vergleich zu anderen Dokumenten im gleichen Index beschrieben wird.

Anmerkung: Die gleiche Syntax wird für die Suchargumente der Suche mit gespeicherten Prozeduren und die SQL-Tabellenwertfunktion verwendet.

Syntax für Suchargumente

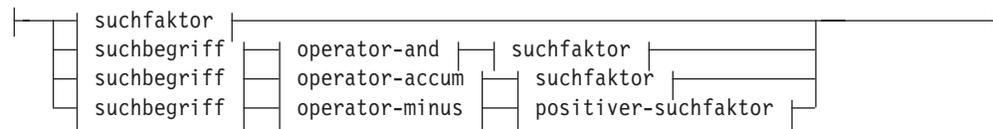
Für die Suche können Argumente mit unterschiedlicher Syntax verwendet werden.



boolescher-suchausdruck:



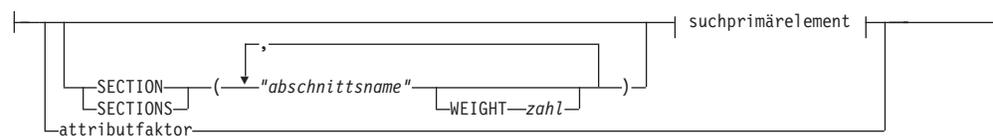
suchbegriff:



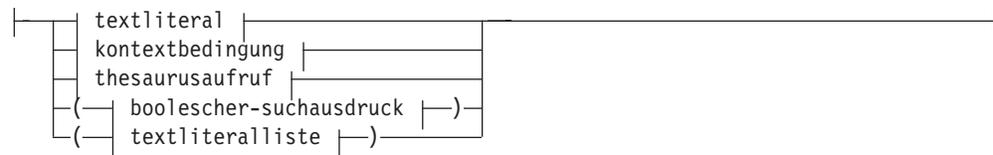
suchfaktor:



positiver-suchfaktor:



suchprimärelement:



operator-and:



operator-or:



operator-accum:



operator-minus:

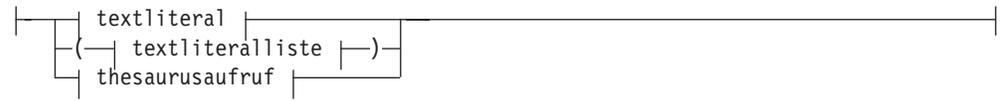


kontextbedingung:





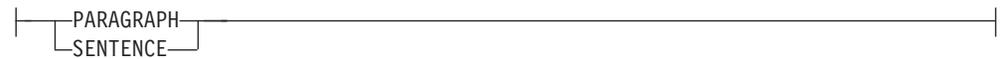
kontextargument:



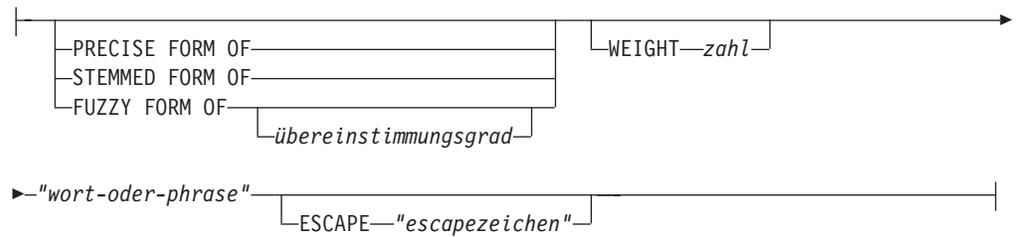
textliteralliste:



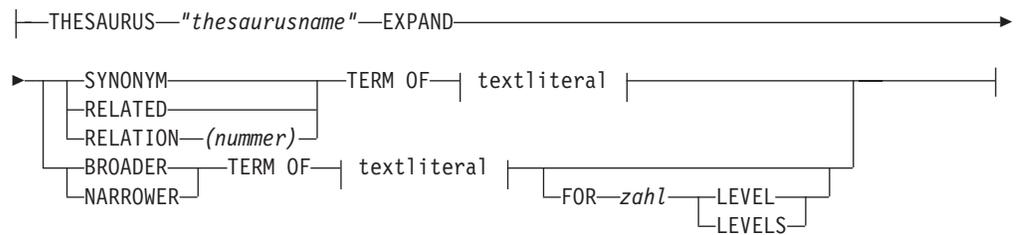
kontexteinheit:



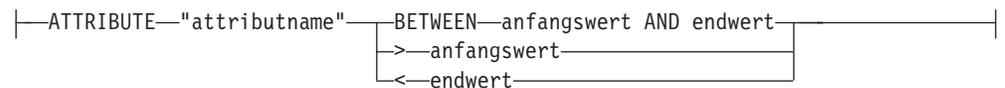
textliteral:



thesaurusaufruf:



attributfaktor:



freitextargument:

|—IS ABOUT—|——"wort-oder-phrase"——|
|——Sprache——| |——ESCAPE—"escapezeichen"——|

Beispiele

Beispiele finden Sie in Kapitel 42, „Angaben von SQL-Suchargumenten“, auf Seite 137.

Suchparameter

In diesem Abschnitt sind die verschiedenen Parametertypen, die bei der Suche verwendet werden können, sowie die zugehörigen Beschreibungen aufgeführt.

Parameter

RESULT LIMIT *zahl*

Ein Schlüsselwort, das die maximale Anzahl von Ergebnissen angibt, die durch die Volltextsuche zurückgegeben werden sollen.

Der Suchparameter RESULT LIMIT sollte zusammen mit der Funktion SCORE verwendet werden, um sicherzustellen, dass die zurückgegebenen Ergebnisse in der Quote berücksichtigt werden und nur die besten übereinstimmenden Ergebnisse verarbeitet werden.

EXPANSION LIMIT *zahl*

Ein Schlüsselwort, das die maximale Anzahl an Begriffen angibt, um die ein Platzhalterzeichenbegriff für die Suche erweitert werden kann. Das Schlüsselwort gibt z. B. an, wie oft Sie den Suchbegriff 'a*' erweitern können. Wenn Ihr Index einen sehr großen Umfang hat und Sie viele Platzhalterbegriffe verwenden, müssen Sie den Wert dieses Schlüsselworts anpassen, wenn Sie eine größere Ergebnismenge abrufen möchten. Die Erweiterungsreihenfolge ist von der internen Organisation des Textindex abhängig und kann nicht vorbestimmt werden. Wenn Ihr Platzhalterausdruck zu allgemein ist und auf mehr Suchbegriffe erweitert werden kann als durch 'EXPANSION LIMIT' angegeben ist, gibt die Suche in einem Fehler zurück, dass das Suchergebnis aufgrund der Überschreitung des Grenzwerts abgeschnitten wurde.

STOP SEARCH AFTER *zahl* DOCUMENT | DOCUMENTS

Ein Schlüsselwort, das einen Schwellenwert für die Suche angibt. Die Suche wird gestoppt, wenn die angegebene Anzahl von Dokumenten während der Suche erreicht und ein Zwischenergebnis zurückgegeben wird. Ein niedrigerer Wert erhöht die Suchleistung, kann jedoch zu weniger Ergebnissen führen und Dokumente mit einem potenziell hohen Rang auslassen.

Beachten Sie, dass es keinen Standardwert gibt und der Wert *zahl* eine positive ganze Zahl sein muss.

boolescher-suchausdruck

Die Suchbegriffe und Suchfaktoren können mit den Booleschen Operatoren NOT, AND, OR, ACCUM und MINUS entsprechend den Syntaxdiagrammen kombiniert werden. Die Operatoren haben die folgende Vorrangfolge (Operatoren mit der höchsten Priorität zuerst): NOT > MINUS = ACCUM = AND > OR. Dies wird durch das folgende Beispiel veranschaulicht:

"Pilot" MINUS "Passagier" & "Fahrzeug" | "Verkehr" & "öffentlich"

Diese Angabe wird wie folgt ausgewertet:

```
(("Pilot" MINUS "Passagier") & ("Fahrzeug")) | ("Verkehr" & "öffentlich")
```

Der Operator ACCUM ergibt den Wert wahr, wenn eines der Booleschen Argumente wahr ist (was mit dem Operator OR vergleichbar ist). Der Rangfolgewert wird durch Akkumulieren der Rangfolgewerte aus beiden Operanden berechnet. Der Operator ACCUM besitzt die gleiche Bindungswalenz (Vorrang) wie AND. Der Operator MINUS ergibt den Wert wahr, wenn der linke Operand als wahr ausgewertet wird. Der Rangordnungswert wird ermittelt, indem vom Rangordnungswert des linken Operanden ein Abzugsbetrag subtrahiert wird, wenn der rechte Operand als wahr ausgewertet wird.

suchprimärelement

Ein Suchprimärelement, das aus einer Textliteralliste besteht, wird als wahr ausgewertet, wenn eines der Textliterate im angegebenen Abschnitt des Dokuments/im Dokument gefunden wird. Ein Suchprimärelement, das aus einem Thesaurusaufruf besteht, wird als wahr ausgewertet, wenn eines der erweiterten Textliterate im angegebenen Abschnitt des Dokuments/im Dokument gefunden wird.

SECTION | SECTIONS *abschnittsname*

Ein Schlüsselwort, das einen oder mehrere Abschnitte in einem strukturierten Dokument angibt, auf die die Suche begrenzt werden soll. Der Abschnittsname muss in einer Modelldatei angegeben sein, die bei der Indexerstellung angegeben wird, oder er muss in XPath-Schreibweise ausgedrückt werden.

Bei Abschnittsnamen wird die Groß-/Kleinschreibung unterschieden. Stellen Sie sicher, dass die Groß-/Kleinschreibung des Abschnittsnamens in der Modelldatei und in der Abfrage identisch ist.

Dieses Modell beschreibt die Struktur von Dokumenten, die identifizierbare Abschnitte enthalten, sodass der Inhalt dieser Abschnitte einzeln durchsucht werden kann. Abschnittsnamen können nicht durch Platzhalterzeichen maskiert werden. Der *positiver-suchfaktor* in der Klausel SECTION wird als wahr ausgewertet, wenn das Suchprimärelement in einem der angegebenen Abschnitte gefunden wird.

Abschnittsnamen sind keine gültigen XPath-Ausdrücke, die bei der Abfrageausführung ausgewertet werden. Wenn keine Modelldatei verwendet wird, werden die Standardabschnittsnamen in XPath-Schreibweise ausgedrückt. Der Ausdruck für den absoluten Pfad für das Element (z. B. /father/child/grandchild) wird als Name für die Abschnittsangabe verwendet. Vollständige XPath-Ausdrücke werden nicht als Abschnittsnamen unterstützt.

kontextargument IN SAME kontexteinheit AS kontextargument AND kontextargument ...

Diese Bedingung gibt Ihnen die Möglichkeit, nach einer Kombination von Textliteralen zu suchen, die im gleichen Absatz oder im gleichen Satz vorkommen. Kontextargumente sind immer äquivalent zu Textliterallisten, und eine Thesauruserweiterung kann dazu verwendet werden, ein Textliteral zu einer solchen Liste zu erweitern.

Die Bedingung wird als wahr ausgewertet, wenn es eine Kontexteinheit (Absatz oder Satz) im Dokument gibt, die mindestens eines der Textliterate jedes erweiterten Kontextarguments enthält. Dies wird durch das folgende Beispiel veranschaulicht:

("a","b") IN SAME PARAGRAPH AS ("c","d")
AND THESAURUS "t1" EXPAND SYNONYM TERM OF "e".

Angenommen, e1 und e2 sind Synonyme von e, dann würden folgende Absätze den Bedingungen entsprechen:

".. a c e .." , ".. a c e1.." , "a c e2..",
".. a d e .." , ".. a d e1.." , "a d e2..",
".. b c e .." , ".. b c e1.." , "b c e2..",
".. b d e .." , ".. b d e1.." , "b d e2..".

PRECISE FORM OF

Ein Schlüsselwort, dass bewirkt, dass das Wort (bzw. jedes Wort in der Phrase) nach **PRECISE FORM OF** genau in der angegebenen Form gesucht wird. Bei dieser Form der Suche wird die Groß-/Kleinschreibung unterschieden. Wenn Sie zum Beispiel nach dem Wort *mouse* suchen, wird das Wort "Mouse" nicht gefunden.

Für diesen Parameter ist es erforderlich, dass der Indexkonfigurationsparameter zur Beachtung der Groß-/Kleinschreibung auf "Ja" gesetzt ist. Wenn der Index erstellt wurde, kann diese Konfigurationseinstellung nicht mehr geändert werden.

STEMMED FORM OF

Mit diesem Schlüsselwort wird das Wort (oder die einzelnen Wörter in der Phrase), das auf **STEMMED FORM OF** folgt, auf dessen Wortstamm zurückgeführt, bevor die Suche ausgeführt wird. Bei dieser Art der Suche wird die Groß-/Kleinschreibung nicht beachtet. Wenn Sie zum Beispiel nach dem Wort *mouse* suchen, wird das Wort "Mouse" gefunden.

Die Art und Weise, wie Wörter auf ihren Stamm zurückgeführt werden, hängt von der Sprache ab. Gegenwärtig wird diese Funktion nur für englische Wortstämme unterstützt, wobei das Wort die regulären Inflexionsendungen aufweisen muss.

FUZZY FORM OF

Mit diesem Schlüsselwort kann eine Suche nach groben Übereinstimmungen durchgeführt werden, d. h. eine Suche nach Begriffen, die eine dem Suchbegriff ähnliche Schreibweise besitzen. Dies ist besonders in Fällen nützlich, in denen Sie Dokumente durchsuchen, die von Programmen für optische Zeichenerkennung (OCR, Optical Character Recognition) erstellt worden sind. Derartige Dokumente enthalten häufig fehlerhaft geschriebene Wörter. Das Wort *Ökonomie* wird von einem OCR-Programm z. B. möglicherweise als *Ökononie* erkannt. Beachten Sie, dass erfolgreiche Übereinstimmungen nur für Wörter in einem Dokument zurückgegeben werden, bei denen die ersten drei Buchstaben übereinstimmen. Im oben stehenden Beispiel ist *Ekonomie* keine Übereinstimmung. Eine Suche nach groben Übereinstimmungen kann nicht verwendet werden, wenn ein Wort im Suchatom ein Platzhalterzeichen enthält.

übereinstimmungsgrad

Eine ganze Zahl zwischen 1 und 100, die den Grad der Ähnlichkeit angibt. 100 bedeutet eine 'exakte Übereinstimmung', während 60 bereits als sehr 'grobe Annäherung' betrachtet wird. Je grober die Vergleichsebene ist, desto länger dauert die Suche, da mehr Dokumente den Bedingungen der Suche entsprechen. Der Standardwert für die Vergleichsebene ist 70.

WEIGHT *zahl*

Ordnet einem Textliteral einen Gewichtungswert zu, um die Standardquote

zu ändern. Die zulässigen Gewichtungswerte sind ganze Zahlen zwischen 0 (die niedrigste Quotengewichtung) und 1000 (die höchste Quotengewichtung). Der Standardwert ist 100.

wort_oder_phrase

Ein Wort oder eine Phrase, nach der gesucht wird. Die Zeichen, die innerhalb eines Worts verwendet werden können, hängen von der Sprache ab. Ob die Wörter durch Trennzeichen getrennt werden müssen, ist ebenfalls sprachabhängig. Im Englischen und in den meisten anderen Sprachen müssen die einzelnen Wörter in einer Phrase durch ein Leerzeichen getrennt werden.

Wenn eine Zeichenfolge gesucht werden soll, die doppelte Anführungszeichen enthält, geben Sie die doppelten Anführungszeichen zweimal ein. Wenn zum Beispiel der Text "Platzhalterzeichen" verwendet' gesucht werden soll, geben Sie dies wie folgt an:

```
""Platzhalterzeichen"" verwenden"
```

Beachten Sie, dass im Beispiel nur nach einem Paar von Anführungszeichen gesucht werden kann. Es ist nicht möglich, in einer Folge nach zwei Paaren von Anführungszeichen zu suchen. Für jedes Wort bzw. jede Phrase gilt außerdem eine Maximallänge von 128 Byte.

Platzhalterzeichen

Ein Wort kann die folgenden Platzhalterzeichen enthalten:

_ (Unterstreichungszeichen)

Steht für ein beliebiges Einzelzeichen.

% (Prozentzeichen)

Steht für eine beliebige Anzahl beliebiger Zeichen. Wenn ein Wort aus einem einzelnen Prozentzeichen % besteht, stellt es ein optionales Wort beliebiger Länge dar. Ein Wort kann nicht ausschließlich aus Platzhalterzeichen zusammengesetzt sein, es sei denn, Sie verwenden zur Darstellung eines optionalen Worts ein einzelnes Prozentzeichen (%). Wenn Sie ein Platzhalterzeichen verwenden, können Sie nicht das Schlüsselwort THESAURUS verwenden. Platzhalterzeichen können nicht in Thesaurusabfrageabschnitten verwendet werden. Wenn Sie beides verwenden, sind die Suchergebnisse unvorhersehbar. Platzhalterzeichen können nur auf alphanumerische Zeichen folgen. Platzhalterzeichen können nicht für eine Suche nach grober Übereinstimmung verwendet werden, da sie nur jeweils in ein einzelnes Wort umgesetzt werden.

ESCAPE *escapezeichen*

Ein Zeichen, das das darauf folgende Zeichen als ein zu suchendes Zeichen, und nicht als Platzhalterzeichen kennzeichnet. Wenn zum Beispiel das Dollarzeichen (\$) ein Escapezeichen ist, dann stellen die Ausdrücke \$%, \$_ und \$\$ die Zeichen %, _ bzw. \$ dar. Alle Prozentzeichen (%) und Unterstreichungszeichen (_), denen kein Dollarzeichen (\$) vorangestellt ist, werden als Platzhalterzeichen interpretiert.

Bei der Suche dürfen Sie nur Einzelbyte-Escapezeichen verwenden. Es sind keine Doppelbytezeichen erlaubt.

THESAURUS *thesaurusname*

Ein Schlüsselwort, das zur Angabe des Namens des Thesaurus dient, der zur Erweiterung eines Textliterals verwendet werden soll. Der Thesaurusname ist der Dateiname (ohne Erweiterung) eines Thesaurus, der mithilfe des Thesauruscompilers kompiliert wurde. Der Thesaurus muss sich im

Verzeichnis <betriebsystemabhängig>/sql11b/db2ext/thes befinden. Alternativ kann auch der vollständige Pfad dem Dateinamen vorangestellt werden.

EXPAND *relation*

Gibt die Relation an, die zur Erweiterung des Textliterals mithilfe des Thesaurus verwendet werden soll. Der Thesaurus enthält vordefinierte Relationen, die im Befehl **DB2EXTTH** beschrieben sind. Diese werden mit den folgenden Schlüsselwörtern angegeben:

- **SYNONYM**, eine symmetrische Beziehung, die Äquivalenz ausdrückt
- **RELATED**, eine symmetrische Beziehung, die Zuordnung ausdrückt
- **BROADER**, eine gerichtete hierarchische Beziehung, der angegebene Tiefengrade nachgestellt werden können
- **NARROWER**, eine gerichtete hierarchische Beziehung, der angegebene Tiefengrade nachgestellt werden können

Benutzerdefinierte Relationen können mit der Option **RELATION**(nummer) angegeben werden, die der Relationsdefinition in **DB2TEXTTH** entspricht.

TERM OF *textliteral*

Das Textliteral, dem weitere Suchbegriffe aus dem Thesaurus hinzuzufügen sind.

zahl **LEVELS**

Ein Schlüsselwort, das zur Angabe der Anzahl von Ebenen (Tiefe) von Begriffen im Thesaurus verwendet wird, die verwendet werden sollen, um den Suchbegriff für eine gegebene Relation zu erweitern. Wenn Sie dieses Schlüsselwort nicht angeben, wird der Wert 1 angenommen. Der Wert des Tiefengrades muss eine positive ganze Zahl (**INTEGER**) sein.

ATTRIBUTE *attributname*

Sucht nach Dokumenten, die Attribute enthalten, die der angegebenen Bedingung entsprechen. Der Attributname bezieht sich auf den Namen eines Attributausdrucks im Befehl **CREATE INDEX** oder auf eine Attributdefinition in der Dokumentmodelldatei.

Der Attributfaktor ist nur für Attribute des Typs **DOUBLE** zulässig. Die Genauigkeit des Werts ist für 15 Stellen garantiert. Zahlen, die aus 16 Ziffern und mehr bestehen, werden gerundet. Die Verwendung von Platzhalterzeichen ist in 'attributname', 'anfangswert' und 'endwert' nicht zulässig. Eine Erläuterung finden Sie in den folgenden Ausführungen:

BETWEEN *anfangswert* **AND** *endwert*

Ein Attributfaktor **BETWEEN** wird als wahr ausgewertet, wenn der Wert des Attributs größer als (nicht gleich) *anfangswert* und kleiner als (nicht gleich) *endwert* ist.

>anfangswert

Ein Attributfaktor ">" wird als wahr ausgewertet, wenn der Wert des Attributs größer als (nicht gleich) 'anfangswert' ist.

<endwert

Ein Attributfaktor "<" wird als wahr ausgewertet, wenn der Wert des Attributs kleiner als (nicht gleich) 'endwert' ist.

Wenn der Attributname im Befehl **CREATE INDEX** mit Anführungszeichen angegeben wird oder in einer Modelldatei definiert ist, muss der angegebene Name exakt übereinstimmen. Wenn hingegen keine Anführungszeichen im Befehl **CREATE INDEX** angegeben werden, muss der Attributname in Großbuchstaben angegeben werden.

IS ABOUT *sprache wort_oder_phrase*

Eine Option, mit der Sie ein Freitextsuchargument angeben können. Wenn Sie **IS ABOUT** verwenden, können Sie in einer beliebigen Reihenfolge in einem Dokument nach einer beliebigen Anzahl (nicht notwendigerweise nach allen) Wörtern suchen, die Sie in 'wort_oder_phrase' angeben. Je näher die Begriffe in der Angabe für 'wort_oder_phrase' beieinander liegen und je mehr Begriffe in das Dokument miteinbezogen werden, desto höher ist der für das Dokument zurückgegebene Quotenwert).

Der Parameter für die Sprache ist optional und muss nur für Thailändisch (TH_TH) gesetzt werden, da er hier für Tokenzwecke erforderlich ist, und für Türkisch (TR_TR), da er hier für die ordnungsgemäße Zuordnung von Groß- und Kleinschreibung erforderlich ist.

Beachten Sie, dass **IS ABOUT** nur von Nutzen ist, wenn Dokumentquotenwerte angefordert werden und die Suchergebnisse nach Quotenwerten sortiert werden.

Kapitel 60. Skalare SQL-Suchfunktion und SQL-Tabellenwertfunktion

Net Search Extender stellt skalare SQL-Suchfunktionen und eine SQL-Tabellenwertfunktion zum Durchsuchen von Textdokumenten bereit, die in DB2 gespeichert sind.

In diesem Abschnitt werden die folgenden SQL-Suchfunktionen beschrieben.

Suchfunktion	Zweck
„CONTAINS (Skalarfunktion)“	Sucht einen Text in einem bestimmten Dokument.
„NUMBEROFMATCHES (Skalarfunktion)“ auf Seite 268	Sucht und gibt die Anzahl der gefundenen Übereinstimmungen zurück.
„SCORE (Skalarfunktion)“ auf Seite 268	Sucht und gibt den Quotenwert eines gefundenen Textdokuments zurück.
„Befehl DB2EXT.TEXTSEARCH“ auf Seite 269	Die SQL-Tabellenwertfunktion gibt eine Tabelle mit Primärschlüsseln zurück, eine Anzahl von Übereinstimmungen und Quotenwerte.
„DB2EXT.HIGHLIGHT“ auf Seite 272	Dient zur Ermittlung von Informationen dazu, warum ein Dokument als Suchergebnis in Frage kommt.

CONTAINS (Skalarfunktion)

Sucht nach Text in einem Textdokument, das durch Net Search Extender indiziert wurde. Die Funktion liefert den ganzzahligen Wert (INTEGER) 1, wenn das Dokument den gesuchten Text bzw. eine beliebige, im Suchargument angegebene Relation enthält. Anderenfalls liefert sie den Wert 0.

Funktionssyntax

►►—CONTAINS—(—*spaltenname*—,—*suchargument*—)—————►►

Funktionsparameter

Funktionsparameter für CONTAINS.

spaltenname

Der Name einer Tabellenspalte. Die Spalte muss einen zugeordneten Textindex besitzen. Textindizes können Sie mit dem Verwaltungsbefehl **DB2TEXT CREATE INDEX** erstellen.

suchargument

Eine Zeichenfolge des Typs VARCHAR, die die zu suchenden Begriffe enthält.

Anmerkung: Die Abfrage mit CONTAINS können Sie nicht bei einem Textindex für eine Sicht verwenden.

NUMBEROFMATCHES (Skalarfunktion)

Durchsucht Textdokumente und gibt einen ganzzahligen Wert (INTEGER) zurück, der die Anzahl der gefundenen Übereinstimmungen pro Dokument angibt.

Funktionssyntax

►—NUMBEROFMATCHES—(—*spaltenname*—,—*suchargument*—)—————►

Funktionsparameter

spaltenname

Der Name einer Tabellenspalte. Die Spalte muss einen zugeordneten Textindex besitzen. Textindizes können Sie mit dem Verwaltungsbefehl **DB2TEXT CREATE INDEX** erstellen.

suchargument

Eine Zeichenfolge des Typs VARCHAR, die die zu suchenden Begriffe enthält.

Anmerkung: Die Abfrage mit NUMBEROFMATCHES können Sie nicht bei einem Textindex für eine Sicht verwenden.

SCORE (Skalarfunktion)

Durchsucht Textdokumente und gibt einen Quotenwert für jedes gefundene Dokument zurück, der angibt, wie gut das gefundene Dokument durch das Suchargument beschrieben wird.

Die Funktion SCORE liefert einen DOUBLE-Wert. Mit zunehmender Häufigkeit des Vorkommens des Suchbegriffs in einem Dokument erhöht sich die Quote des Dokuments.

Funktionssyntax

►—SCORE—(—*spaltenname*—,—*suchargument*—)—————►

Funktionsparameter

spaltenname

Der Name einer Spalte. Die Spalte muss einen zugeordneten Textindex besitzen. Textindizes können Sie mit dem Verwaltungsbefehl **DB2TEXT CREATE INDEX** erstellen.

suchargument

Eine Zeichenfolge des Typs VARCHAR, die die zu suchenden Begriffe enthält.

Anmerkung: Die Abfrage mit SCORE können Sie nicht bei einem Textindex für eine Sicht verwenden. Die nach Quote zurückgegebenen Werte sind nur dann aussagefähig, wenn sie mit anderen Werten verglichen werden, die vom gleichen Index abgerufen werden.

Befehl DB2EXT.TEXTSEARCH

Neben der Suche mit gespeicherten Prozeduren und den skalaren SQL-Suchfunktionen stellt Net Search Extender zwei SQL-Tabellenwertfunktionen bereit, die der gespeicherten Prozedur sehr ähnlich sind.

Zweck

Beide Tabellenwertfunktionen heißen `db2ext.textsearch`. Der einzige Unterschied besteht darin, dass eine Funktion die Funktion `HIGHLIGHT` unterstützt und zwei zusätzliche Parameter, `numberOfHits` und `hitInformation`, besitzt.

Beide Tabellenwertfunktionen geben die Ergebnisse der Benutzertabellen nach den Sortierkriterien sortiert zurück, die in dem Parameter **INITIAL SEARCH RESULT ORDER** des Befehls **CREATE INDEX** definiert wurden. Wenn die SQL-Abfrageanweisung mit der Tabellenwertfunktion am Ende der Anweisung über einen Join verfügt (z. B. `where T.primkey = S.key`), ist die Reihenfolge der Ergebniszeilen von der Joinmethode und nicht von der im Befehl **CREATE INDEX** definierten Reihenfolge abhängig.

- Sie können die Tabellenwertfunktion nicht für Tabellen mit einem zusammengesetzten Primärschlüssel verwenden.
- Die Tabellenwertfunktion kann in einer Umgebung mit partitionierten Datenbanken nur dann verwendet werden, wenn die Benutzertabelle in einem Tabellenbereich gespeichert wird, der sich auf einer einzelnen Partition befindet. Außerdem müssen Sie über die Umgebungsvariable **DB2NODE** sicherstellen, dass Sie die Verbindung zum richtigen Knoten herstellen.

1. `db2ext.textsearch` ohne Unterstützung für `HIGHLIGHT`

```
db2ext.textSearch
(
    query           VARCHAR(4096),
    indexSchema     VARCHAR(128),
    indexName       VARCHAR(128),
    resultFirstRow  INTEGER,
    resultNumberRows INTEGER,
    primKeyBinding  <unterstützte typen>, // selber Typ wie Primärschlüssel
)

return table
(
    primKey         <unterstützte typen>, // selber Typ wie Primärschlüssel
    numberOfMatches INTEGER,
    score           DOUBLE,
    totalNbResults INTEGER
)
```

2. `db2ext.textsearch` mit Unterstützung für `HIGHLIGHT`

```
db2ext.textSearch
(
    query           VARCHAR(4096),
    indexSchema     VARCHAR(128),
    indexName       VARCHAR(128),
    resultFirstRow  INTEGER,
    resultNumberRows INTEGER,
    primKeyBinding  <unterstützte typen>, // selber Typ wie Primärschlüssel
    numberOfHits    INTEGER
)
```

```

return table
(
  primKey          <unterstützte typen>, // selber Typ wie Primärschlüssel
  numberOfMatches  INTEGER,
  score            DOUBLE,
  totalNbResults   INTEGER
  hitInformation   BLOB(20K)
)

```

Funktionsparameter

Es gibt folgende Eingabeparameter.

query Weitere Informationen finden Sie in Kapitel 59, „Syntax für Suchargumente“, auf Seite 257.

indexSchema, indexName

Gibt den zu durchsuchenden Index an.

resultFirstRow

Die Ergebnisliste der Abfrage (query) wird in Teilen zurückgegeben. Dieser Parameter beschreibt, welche Zeile der Abfrageergebnisliste die erste Zeile ist, die in die Ergebnistabelle der Tabellenwertfunktion eingefügt werden soll. Der Wert muss ≥ 0 sein.

Beachten Sie, dass die Ziffer 0 die erste Zeile in der Abfrageergebnisliste kennzeichnet.

resultNumberRows

Dieser Parameter beschreibt, wie viele Zeilen der Abfrageergebnisliste in die Ergebnistabelle der Tabellenwertfunktion eingefügt werden sollen, wobei 0 bedeutet, dass alle Ergebnisse zurückzugeben sind.

Beachten Sie, dass sich dieser Wert von dem Abfrageparameter zur Begrenzung des Ergebnisses unterscheidet, der die maximale Größe der Abfrageergebnisliste festlegt.

primaryKeyBinding

Der Typ dieses Parameters legt den Typ des Ausgabeparameters **primaryKey** fest. Wenn der Textindex für eine Basistabelle mit einem Primärschlüsseltyp <typ1> erstellt wurde, muss der Parameter **primaryKeyBinding** ebenfalls vom Typ <typ1> sein.

Darüber hinaus bestimmt der Parameter den Geltungsbereich der Textsuche. Wenn der Parameter **primaryKeyBinding** auf NULL ("CAST(NULL as <typ1>)" gesetzt wird, erstreckt sich der Geltungsbereich der Suche auf alle im Index gespeicherten Dokumente. Alternativ können Sie die Suche auf die Dokumente beschränken, an die sie durch den Parameter **primaryKeyBinding** gebunden wird.

Wenn **primaryKeyBinding** zum Beispiel auf den Wert CAST(5 as BIGINT) gesetzt wird, beschränken Sie die Suche auf ein einziges Dokument mit dem Primärschlüsselwert '5' des Typs BIGINT.

Beachten Sie, dass nur einspaltige Primärschlüssel der folgenden Typen unterstützt werden: SMALLINT, INTEGER, BIGINT, REAL, DOUBLE, VARCHAR FOR BIT DATA, DATE, TIME und TIMESTAMP.

numberOfhits

Mit dieser Option wird die Anzahl an Begriffen angegeben, die mit der Hervorhebungsfunktion `db2ext.highlight` hervorgehoben werden. Bei der

Angabe von 0 werden alle Treffer bis zu einem Maximum von 1.100 Treffern hervorgehoben. Dieser Prozess kann zeitaufwendig sein.

Funktionsparameter

Die folgenden Rückgabewerte werden in einer temporären Tabelle gespeichert, die mit Ihrer Benutzertabelle verknüpft werden muss, wenn weitere Ergebnisse angefordert werden. Beachten Sie, dass die Werte für NUMBEROFMATCHES, SCORE, TOTALNUMBEROFRESULTS und HITINFORMATION nur berechnet werden, wenn sie in Ihrer SELECT-Anweisung angefordert werden.

primKey

Der Primärschlüssel des gefundenen Dokuments.

numberofmatches

NUMBEROFMATCHES ist ein INTEGER-Wert, der die Anzahl von Übereinstimmungen angibt, die für die einzelnen Dokumente ermittelt wurden.

score SCORE liefert einen DOUBLE-Wert. Je häufiger das Vorkommen des Suchbegriffs in einem Dokument, desto höher der SCORE-Wert (Quote) des Dokuments.

totalNumberOfResults

Die Abfrageergebnisliste gibt an, wie viele Ergebnisse gefunden wurden. Beachten Sie, dass jede Zeile den gleichen Wert aufweist.

Beachten Sie auch, dass diese Zahl nicht mehr zuverlässig ist, wenn Sie STOP SEARCH AFTER oder RESULT LIMIT zusammen mit der SCORE-Syntax in einer Abfrage verwenden.

hitInformation

Die von db2ext.textsearch zurückgegebenen Trefferinformationen werden für die Verarbeitung der Hervorhebungen benötigt. Derzeit kann dieser Ausgabeparameter Trefferinformationen für etwa 1100 Treffer enthalten. Wenn die Anzahl der Treffer diesen Schwellenwert übersteigt, werden die Trefferinformationen für diese überzähligen Treffer ignoriert.

Beachten Sie, dass dieser Wert nur zurückgegeben wird, wenn Sie einen Wert für numberOfHits angeben.

Syntax

Mithilfe der SQL-Tabellenwertfunktion sind Sie in der Lage, Sichten in gleicher Weise zu durchsuchen, wie dies durch die Suche mit gespeicherten Prozeduren möglich ist. Nur mit der SQL-Tabellenfunktion ist kein gemeinsamer Speicher erforderlich, sodass für den Index kein Cache erforderlich ist, der aktiviert werden muss.

Diese Funktion ist in erster Linie für Benutzer gedacht, die eine SQL-Abfrage innerhalb der Suche mit gespeicherten Prozeduren verwendet haben. Eine Einschränkung besteht jedoch darin, dass nur ein einspaltiger Primärschlüssel für Basistabellen unterstützt wird.

Das folgende Beispiel zeigt den Umgang mit einer Tabelle mit einem mehrspaltigen Primärschlüssel:

```
select s.id from
db2ext.sample s, table (db2ext.textSearch(
    'characteristics',
    'DB2EXT',
    'COMMANDS',
```

```

1,
20,
cast(NULL as INTEGER)) t
where s.id = t.primkey

```

In diesem Beispiel müssen Sie zunächst eine Sicht auf diese Tabelle mit einem eindeutigen einspaltigen Schlüssel und anschließend den Index für diese Sicht erstellen.

Ein Beispiel für die Verwendung der SQL-Tabellenwertfunktion mit der Funktion `db2ext.highlight` finden Sie in „DB2EXT.HIGHLIGHT“.

DB2EXT.HIGHLIGHT

Mit der Funktion 'db2ext.highlight' können Sie Informationen abrufen, mit deren Hilfe Sie anzeigen können, warum ein bestimmtes Dokument als Suchergebnis ausgewählt wurde.

Im Einzelnen können Sie mit dieser Funktion folgende Aktionen durchführen:

- Treffer abrufen
- Treffer und umgebenden Text abrufen
- das Dokument mit benutzerdefinierten Hervorhebungstags zur Kennzeichnung der Treffer abrufen

Beachten Sie, dass die Funktion 'db2ext.highlight' nur zusammen mit der Tabellenwertfunktion 'db2ext.textsearch' verwendet werden kann. Die Tabellenwertfunktion durchsucht den Index und liefert die Ergebnisse, auf die die Funktion `HIGHLIGHT` angewandt werden soll.

Funktionssyntax

```

▶—db2ext.highlight—▶
▶(—dokumentinhalt—,—trefferinformationen—,—informationen_zur_trefferverarbeitung—)▶

```

Funktionsparameter

Es gibt folgende Eingabeparameter:

dokumentinhalt CLOB(100K)

Es werden nur UTF8-Dokumente im TEXT- oder in serialisiertem XML-Format unterstützt. Zum Hervorheben nativ gespeicherter XML-Dokumente müssen die XML-Daten für CLOB mithilfe der SQL-/XML-Funktion `XMLSERIALIZE` serialisiert werden.

Wenn Sie den CLOB-Wert erhöhen möchten, verwenden Sie das „Befehl `DB2EXTTH`“ auf Seite 245.

trefferinformationen BLOB(20K)

Eine Zeichenfolge mit Trefferinformationen. Diese Zeichenfolge wird von der Funktion 'db2ext.textsearch' zurückgegeben, wenn ein Wert für den Parameter `numberOfHits` angegeben wird.

informationen_zur_trefferverarbeitung VARCHAR(1024)

Dieser Parameter ist eine Liste von Paaren der Form `Option=Wert`, die durch ein Komma (,) getrennt sind, wobei jedes Zeichen der Zeichenfolge in " " eingeschlossen ist. Der Parameter bestimmt die Verarbeitung der Her-

vorhebungen im angegebenen Dokument. Wenn keine Option angegeben wird, wird der Inhalt des Originaldokuments unverändert zurückgegeben.

TAGS = ("STRING", "STRING")

Mit dieser Option kann der Benutzer die Tags angeben, die vor und nach einem Treffer in das Dokument einzufügen sind. Wenn Sie diese Option auslassen, werden vor und nach einem Treffer im Dokument keine Tags eingefügt.

WINDOW_NUMBER = INTEGER

Diese Option legt fest, wie viele Teile (oder Fenster) des Dokuments von der Hervorhebungsfunktion zurückgegeben werden sollen. Jedes Fenster enthält einen oder mehrere Treffer. Der erste Treffer in jedem Fenster legt den Teil des Dokuments fest, der an den Benutzer zurückgegeben wird. Diese Treffer können von Text umgeben sein oder nicht.

Wenn diese Option ausgelassen wird, wird 0 als Standardwert verwendet, und das gesamte Dokument wird mit Start- und Endtags (falls definiert) zurückgegeben. In diesem Fall wird die Option WINDOW_SIZE ignoriert.

WINDOW_SIZE = INTEGER

Diese Option gibt die empfohlene Fenstergröße in Byte an. Die tatsächliche Größe kann abhängig von der Anzahl der Treffer, der Länge der Treffer und der Größe der Start- und Endtags variieren. Wenn diese Option ausgelassen wird, ist 0 der Standardwert, und nur Treffer ohne umgebenden Text werden zurückgegeben.

WINDOW_SEPARATOR = "STRING"

Diese Option gibt den Tag an, der zur Trennung eines Fensters vom nächsten Fenster verwendet wird. Wenn die Option nicht angegeben wird, verwendet das System den Standardwert '...'.

FORMAT = "STRING"

Diese Option gibt das Format des Dokuments an. Gültige Werte sind XML oder TEXT. Wenn diese Option ausgelassen wird, wird TEXT als Standardwert verwendet. Stellen Sie sicher, dass der Formatwert mit dem bei der Indexierung angegebenen Wert übereinstimmt.

MODEL_NAME = "STRING"

Mit dieser Option wird der Modellname angegeben, der dem angegebenen XML-Dokument zugeordnet ist. Beachten Sie hierbei, dass diese Option zu einer Fehlerbedingung führt, wenn als Format (FORMAT) die Einstellung TEXT verwendet wird.

SECTIONS = ("abschnittsname1", ..., "abschnittsnameN")

Bei XML-Dokumenten kann die Hervorhebung auf die relevanten Abschnitte beschränkt werden. Sie können z. B. in der Modelldatei definiert werden. Bei Angabe der Abschnitte werden die Namen durch Kommas getrennt. Wenn diese Option ausgelassen wird, wird die Hervorhebungsfunktion auf das gesamte XML-Dokument angewendet. Beachten Sie, dass diese Option ignoriert wird, wenn die Option FORMAT gleich TEXT ist.

Die in DB2EXT.HIGHLIGHT verwendete Abschnittsspezifikation ("abschnittsname1",..., "abschnittsnameN") muss mit der für die Funktion DB2EXT.TEXTSEARCH verwendeten übereinstimmen.

Funktionsparameter

Es gibt folgende Rückgabeparameter.

CLOB(200K)

Die Funktion HIGHLIGHT gibt einen CLOB-Wert zurück, der die von der Funktion HIGHLIGHT geänderten Dokumententeile enthält.

Syntax

Das folgende Beispiel zeigt die Verwendung der Funktion HIGHLIGHT:

```
select p.id,
       p.title,
       db2ext.highlight(p.content,
                       t.hitinformation,
                       'TAGS = ("<bf>", "</bf>"),
                       WINDOW_NUMBER = 5,
                       WINDOW_SIZE = 200,
                       WINDOW_SEPARATOR = "...",
                       FORMAT = "XML",
                       SECTIONS = ("abschnittsname1", "abschnittsname2"))
FROM patent p, table (db2ext.textsearch(
    "relationale Datenbanksysteme",
    'DB2EXT',
    'TI_FOR_CONTENT',
    0,
    20,
    CAST(NULL as BIGINT),
    15)) t
WHERE p.id = t.primkey
```

Die Verwendung von Dokumenten mit mehr als 100 KB führt zum Abbruch der SQL-Abfrage und zur Ausgabe eines SQL-Fehlers (SQL1476N und SQL-Fehler -433). Zur Vermeidung dieses Problems können Sie mit dem Befehl db2exthl die zulässige Dokumentinhaltsgröße erhöhen.

Anmerkung: Sonderzeichen wie z. B. das Zeilenschaltungszeichen werden unverändert zurückgegeben.

Wenn Sie nativ gespeicherte XML-Dokumente hervorheben, ist eine serielle Verarbeitung dieser XML-Dokumente für CLOB erforderlich, bevor eine Übergabe an die Tabellenwertfunktion HIGHLIGHT möglich ist. Das folgende Beispiel zeigt die Verwendung der Funktion HIGHLIGHT für nativ gespeicherte XML-Dokumente mit der SQL-/XML-Funktion XMLSERIALIZE: Der Patentinhalt des folgenden Beispiels wird als native XML gespeichert. Beachten Sie, dass FORMAT="XML" ebenfalls angegeben wird:

```
select p.id,
       p.title,
       db2ext.highlight(XMLSERIALIZE(p.content AS CLOB(100K)),
                       t.hitinformation,
                       'TAGS = ("<bf>", "</bf>"),
                       FORMAT = "XML",
                       SECTIONS = ("abschnittsname1", "abschnittsname2"))
FROM patent p, table (db2ext.textsearch(
    "xml database systems",
    'DB2EXT', 'TI_FOR_XML',
    0,
    20,
```

```
CAST(NULL as BIGINT),  
15)) t
```

```
WHERE p.id = t.primkey
```

Einschränkungen

- Es werden nur XML- und unstrukturierte Textdokumente unterstützt.
- Nur UTF8-Datenbanken werden unterstützt. Bei binären Dokumenten müssen Sie sicherstellen, dass die Dokumente in UTF8 codiert sind.
- Thailändische Dokumente werden nicht unterstützt.
- Wenn das bei der Indexierung verwendete Dokumentformat von dem bei der Abfrage verwendeten Dokumentformat abweicht, gibt die Funktion HIGHLIGHT unvorhergesehene Ergebnisse zurück. Dies ist insbesondere in solchen Fällen zutreffend, in denen während der Indexierung eine Umsetzungsfunktion verwendet wurde, um den Text zu erhalten. Jegliche Änderung an der Umsetzungsfunktion, die zwischen dem Zeitpunkt der Indexierung und dem Zeitpunkt der Übergabe der Suchanfrage vorgenommen wurde und Auswirkungen auf die Position von Texttoken in der Ausgabe hat, führt dazu, dass die Ergebnisse der Hervorhebungsfunktion ungültig sind.
- Es werden nur Treffer hervorgehoben, die in den Textteilen eines Dokuments gefunden werden.
- Die Tabellenwertfunktion HIGHLIGHT kann nur in Verbindung mit der Funktion 'db2ext.textsearch' verwendet werden.
- Zeichenfolgewerte dürfen nicht das Zeichen " enthalten.

Kapitel 61. Funktion zur Suche mit gespeicherten Prozeduren

Net Search Extender stellt eine Suche mit gespeicherten Prozeduren zur Rückgabe vordefinierter Ergebnistabellen bereit. Die Ergebnistabelle wird im Cachetabellenabschnitt bei der Indexerstellung angegeben.

Verwenden Sie die Suche mit gespeicherten Prozeduren zur Rückgabe einer kleinen Zahl von Ergebnissen in einer bestimmten Reihenfolge.

Ein Beispiel wäre eine Webanwendung, bei der die ersten 20 Zeilen mit den besten Ergebnissen zurückgegeben werden; allerdings können auch die übrigen Ergebnisse in Gruppen von je 20 Zeilen zurückgegeben werden.

Anmerkung: Die Funktion für gespeicherte Prozeduren kann in einer Umgebung mit partitionierten Datenbanken nur dann verwendet werden, wenn die Benutzertabelle in einem Tabellenbereich gespeichert wird, der sich auf einer einzelnen Partition befindet.

Außerdem müssen Sie über die Umgebungsvariable **DB2NODE** sicherstellen, dass Sie die Verbindung zur richtigen Partition herstellen.

DB2EXT.TEXTSEARCH für die Suche mit gespeicherten Prozeduren

Die Spalten in der Ergebnismenge, die von der gespeicherten Prozedur zurückgegeben werden, werden durch die Option **CACHE TABLE** des Befehls **DB2TEXT CREATE INDEX** definiert. Wenn `scoringFlag=1` ist, wird eine Spalte des Typs **DOUBLE** hinzugefügt.

Funktionssyntax

`db2ext.TextSearch(`

IN	query	VARCHAR(4096),
IN	indexSchema	VARCHAR(128),
IN	indexName	VARCHAR(128),
IN	resultFirstRow	INTEGER,
IN	resultNumberRows	INTEGER,
IN	scoringFlag	INTEGER,
IN	searchTermCountsFlag	INTEGER,
OUT	searchTermCounts	VARCHAR(4096),
OUT	totalNumberOfResults	INTEGER)

Funktionsparameter

Es gibt folgende Eingabeparameter.

query Weitere Informationen hierzu finden Sie in Kapitel 59, „Syntax für Suchargumente“, auf Seite 257.

indexSchema, indexName

Gibt den zu durchsuchenden Index an.

resultFirstrow

Die Abfrageergebnisliste wird in Teilen zurückgegeben. Dieser Parameter beschreibt, welche Zeile der Abfrageergebnisliste die erste Zeile ist, die in

die Ergebnismenge der gespeicherten Prozedur eingefügt werden soll. Die erste Zeile in der Abfrageergebnisliste ist durch die Ziffer 0 gekennzeichnet.

resultNumberRows

Dieser Parameter beschreibt, wie viele Zeilen der Abfrageergebnisliste in die Ergebnismenge der gespeicherten Prozedur eingefügt werden sollen.

Dieser Parameter ist nicht mit dem Ergebnisbegrenzungsausdruck (RESULT LIMIT) in der Abfrage zu verwechseln, der die maximale Größe der Abfrageergebnisliste festlegt.

Der Wert sollte ≥ 0 sein. Dabei bedeutet 0, dass alle Ergebnisse zurückgegeben sind.

Anmerkung: Wenn eine größere Ergebnismenge angefordert wird, stellen Sie sicher, dass ein Tabellenbereich für temporäre Benutzertabellen verfügbar ist. Wenn keiner verfügbar ist, erstellen Sie einen Tabellenbereich. Im folgenden Beispiel wird ein Tabellenbereich unter einem UNIX-Betriebssystem erstellt:

```
db2 "create user temporary tablespace tempts managed by system
      using ('/work/temptts.ts')"
```

scoringFlag

Der Wert 0 bedeutet, dass keine Quoten ermittelt werden, der Wert 1, dass Quoten ermittelt werden. Wenn die Ermittlung von Quoten erforderlich ist, wird eine zusätzliche Spalte mit den Quotenwerten zurückgegeben, wobei der höchste Wert zuerst zurückgegeben wird.

searchTermCountsFlag

Dieser Parameter steuert die Verarbeitung des Parameters searchTermCounts. Wenn der Parameter **searchTermCountsFlag** den Wert 0 aufweist, wird der Wert des Parameters searchTermCounts nicht berechnet.

Funktionsparameter

Es gibt folgende Ausgabeparameter.

searchTermCounts

Die Anzahl von Vorkommen der einzelnen Abfragesuchbegriffe im Index. Diese Zahlen werden als eine durch Leerzeichen getrennte Liste in der Reihenfolge der Suchbegriffe in der Abfrage zurückgegeben.

Weitere Informationen finden Sie unter **searchTermCountsFlag**.

totalNumberOfResults

Die Gesamtzahl von Ergebnissen, die in der Abfrageergebnisliste zu finden sind.

Beachten Sie auch, dass diese Zahl nicht mehr zuverlässig ist, wenn Sie in einer Abfrage STOP SEARCH AFTER oder RESULT LIMIT zusammen mit der scoringFlag-Syntax verwenden.

Syntax

Die Spalten in der Ergebnismenge, die von der gespeicherten Prozedur zurückgegeben werden, werden durch die Option **CACHE TABLE** des Befehls **DB2TEXT CREATE INDEX** definiert. Wenn scoringFlag=1 ist, wird eine Spalte des Typs DOUBLE hinzugefügt. Diese Spalte enthält den SCORE-Wert.

Mit den folgenden Optionen können Sie die Leistung einer zweiten Abfrage verbessern, deren Abfragezeichenfolge mit der der ersten Abfrage identisch ist. Beachten Sie, dass dazu ein anderes Cursorfenster ohne `totalNumberOfResults` erforderlich ist:

- Wenn keine Quotenermittlung erforderlich ist, fügen Sie folgende Syntax hinzu: `STOP SEARCH AFTER x DOCUMENTS`; dabei entspricht `x` der Summe von `resultFirstRow + resultNumberRows`.
- Wenn eine Quotenermittlung erforderlich ist, fügen Sie folgende Syntax hinzu: `STOP SEARCH AFTER y DOCUMENTS`; dabei entspricht `y` dem Wert von `totalNumberOfResults` in der ersten Abfrage.

Um sicherzustellen, dass Sie für die Suche eine Verbindung zum richtigen Knoten herstellen, muss unter Umständen die Umgebungsvariable **DB2NODE** gesetzt werden.

Verwenden Sie unter UNIX den folgenden Befehl:

```
export DB2NODE=<no>
```

Hierbei ist zu beachten, dass alle physischen Knoten über eine synchronisierte Uhrzeit verfügen müssen.

Verwenden Sie unter Windows den folgenden Befehl:

```
set DB2NODE= <no>
```

<p>Anmerkung: Eine abgeschirmte Benutzer-ID, die ungleich der Instanzeigner-ID ist, funktioniert nicht mit partitionierten Datenbanken.</p>
--

Kapitel 62. Windows-Systemfehler

Im Folgenden sehen Sie eine Liste mit Windows-Systemfehlern:

Systemfehler

- 1 Fehlerhafte Funktion.
- 2 Das System kann die angegebene Datei nicht finden.
- 3 Das System kann den angegebenen Pfad nicht finden.
- 4 Das System kann die Datei nicht öffnen.
- 5 Zugriff verweigert.
- 6 Die Zugriffsnummer ist nicht definiert.
- 8 Nicht genügend Arbeitsspeicher verfügbar, um diesen Befehl auszuführen.
- 14 Es steht nicht genügend Speicherplatz zum Beenden dieser Operation zur Verfügung.
- 15 Das System kann das angegebene Laufwerk nicht finden.
- 29 Das System kann nicht auf das angegebene Gerät schreiben.
- 30 Das System kann nicht vom angegebenen Gerät lesen.
- 32 Der Prozess kann auf die Datei nicht zugreifen, da diese momentan von einem anderen Prozess verwendet wird.
- 36 Zu viele Dateien zur gemeinsamen Verwendung geöffnet.
- 38 Das Ende der Datei wurde erreicht.
- 39 Der Datenträger ist voll.
- 80 Die Datei existiert.
- 82 Das Verzeichnis oder die Datei kann nicht erstellt werden.
- 100 Es kann keine weitere Systemsemaphore erstellt werden.
- 101 Die exklusive Semaphore gehört einem anderen Prozess.
- 102 Die Semaphore ist gesetzt und kann nicht geschlossen werden.
- 103 Die Semaphore kann nicht erneut gesetzt werden.
- 104 Zur Zeit einer Unterbrechung (Interrupt) kann keine exklusive Semaphore angefordert werden.
- 105 Das bisher bestehende Besitzrecht dieser Semaphore ist beendet.
- 110 Das System kann das angegebene Gerät oder die Datei nicht öffnen.
- 111 Der Dateiname ist zu lang.
- 112 Es steht nicht genügend Speicherplatz auf dem Datenträger zur Verfügung.
- 121 Das Zeitlimit für die Semaphore wurde erreicht.
- 126 Das angegebene Modul wurde nicht gefunden.
- 127 Die angegebene Prozedur wurde nicht gefunden.

- 147 Zur Verarbeitung dieses Befehls stehen nicht genügend Ressourcen zur Verfügung.
- 155 Es kann kein weiterer Thread erstellt werden.
- 161 Der angegebene Pfadname ist ungültig.
- 164 Es können keine weiteren Threads im System angelegt werden.
- 170 Die angeforderte Ressource wird bereits verwendet.
- 183 Eine Datei kann nicht erstellt werden, wenn sie bereits existiert.
- 187 Der angegebene System-Semaphorename wurde nicht gefunden.
- 206 Der Dateiname oder die Erweiterung ist zu lang.
- 267 Unzulässiger Verzeichnisname.
- 288 Es wurde versucht, einen Mutex freizugeben, der nicht im Besitz des Aufrufers war.
- 298 Es wurde zu oft an eine Semaphore gesendet.
- 998 Unzulässiger Zugriff auf einen Speicherbereich.
- 1051 Ein Stoppzeichen wurde an einen Dienst gesendet, von dem andere Dienste abhängen.
- 1052 Die Steuerungsanforderung ist für diesen Dienst nicht zulässig.
- 1053 Der Dienst antwortete nicht rechtzeitig auf die Start- oder Steuerungsanforderung.
- 1054 Für den Dienst konnte kein Thread erstellt werden.
- 1055 Die Dienstdatenbank ist gesperrt.
- 1056 Es wird bereits eine Instanz des Dienstes ausgeführt.
- 1057 Der Kontoname ist ungültig oder nicht vorhanden oder das Kennwort ist für den angegebenen Kontonamen ungültig.
- 1058 Der Dienst kann nicht gestartet werden, weil dieser inaktiviert ist oder weil ihm keine aktivierten Geräte zugeordnet sind.
- 1059 Es wurde eine Ringdienstabhängigkeit angegeben.
- 1060 Der angegebene Dienst ist kein installierter Dienst.
- 1061 Der Dienst kann zur Zeit keine Steuerungsmeldungen annehmen.
- 1062 Der Dienst wurde nicht gestartet.
- 1063 Der Dienstprozess konnte keine Verbindung zum Dienstcontroller herstellen.
- 1064 Bei der Verarbeitung der Steuerungsanforderung ist eine Ausnahme im Dienst aufgetreten.
- 1066 Der Dienst hat einen dienstspezifischen Fehlercode zurückgegeben.
- 1067 Der Prozess wurde unerwartet beendet.
- 1068 Der Abhängigkeitsdienst oder die Abhängigkeitsgruppe konnte nicht gestartet werden.
- 1069 Der Dienst konnte wegen einer fehlerhaften Anmeldung nicht gestartet werden.
- 1070 Der Dienst konnte nicht gestartet werden.

- 1071 Die angegebene Sperre für die Dienstdatenbank ist unzulässig.
- 1072 Der angegebene Dienst wurde zum Löschen markiert.
- 1073 Der angegebene Dienst ist bereits vorhanden.
- 1078 Der Name wird bereits verwendet als Dienstname oder als Dienstanzeigename.
- 1079 Das für diesen Dienst angegebene Konto unterscheidet sich von dem für andere Dienste angegebenen Konto, die in diesem Vorgang ausgeführt werden.
- 1082 Es wurde kein Recoveryprogramm für diesen Dienst konfiguriert.
- 1154 Eine Bibliotheksdatei, die für die Ausführung dieser Anwendung erforderlich ist, ist beschädigt.
- 1219 Die angegebenen Referenzen passen nicht zu einer bestehenden Referenzmenge.
- 1242 Der Dienst ist bereits registriert.
- 1243 Der angegebene Dienst ist nicht vorhanden.
- 1244 Der angeforderte Vorgang konnte nicht ausgeführt werden, da die Echtheit des Benutzers nicht bestätigt wurde.
- 1245 Der angeforderte Vorgang konnte nicht ausgeführt werden, da der Benutzer nicht am Netz angemeldet ist. Der angegebene Dienst ist nicht vorhanden.
- 1392 Die Datei oder das Verzeichnis ist beschädigt und nicht lesbar.
- 1455 Die Auslagerungsdatei ist zu klein, um diesen Vorgang durchzuführen.
- 1793 Das Benutzerkonto ist abgelaufen.

Kapitel 63. Net Search Extender-Informationskataloge

Net Search Extender speichert wichtige Informationen über Standardwerte, Konfigurationen, Textindizes und Formate in Katalogtabellen. Um diese Informationen abzurufen, können Sie einige Sichten abfragen, die für diese Tabellen definiert sind.

Die folgenden Sichten und Tabellen geben Informationen zur aktuellen Konfiguration Ihres Systems wieder:

- Informative Sichten auf Datenbankebene:
 - `db2ext.dbdefaults`
- Informative Sichten auf Indexebene:
 - `db2ext.textindexes`
 - `db2ext.textindexformats`
 - `db2ext.indexconfiguration`
- Tabellensichten für einen Textindex:
 - Ereignissicht
 - Protokolltabellensicht
 - Zwischenspeichertabellensicht

Sichten für Informationen auf Datenbankebene

Die Sicht '`db2ext.dbdefaults`' enthält alle Standardwert für die Net Search Extender-Datenbank.

Die Standardwerte auf der Datenbankebene können geändert werden und sind als Attribut-Wert-Paare in folgender Sicht verfügbar:

```
db2ext.dbdefaults
```

```
db2 select DEFAULTNAME, DEFAULTVALUE from DB2EXT.DBDEFAULTS
```

Tabelle 3. Sicht `db2ext.dbdefaults`

Attribut	Standardwert	Anmerkungen
CCSID	CCSID der Datenbank	Die Standard-CCSID (CCSID = ID für codierten Zeichensatz) für Dokumente. Dieser Wert wird verwendet, wenn im Befehl <code>CREATE INDEX</code> keine CCSID angegeben wird.
FORMAT	TEXT	Das Dokumentstandardformat. Dieses Format wird verwendet, wenn im Befehl <code>CREATE INDEX</code> kein Format angegeben wird.

Tabelle 3. Sicht *db2ext.dbdefaults* (Forts.)

Attribut	Standardwert	Anmerkungen
INDEXDIRECTORY	Siehe Pfadname unter 'Ammerkungen'.	Verzeichnis für Volltextindexdateien. Dieses Verzeichnis wird verwendet, wenn im Befehl CREATE INDEX kein Indexverzeichnis angegeben wird. Bei Linux- und UNIX-Betriebssystemen: <i>INSTHOME</i> /sql1lib/db2ext/indexes. Dabei ist <i>INSTHOME</i> das Ausgangsverzeichnis des Instanzeigners. Bei Windows-Betriebssystemen: <i>INSTPROFDIR</i> \instanzname\db2ext\indexes. Dabei ist <i>INSTPROFDIR</i> das Instanzprofilverzeichnis.
LANGUAGE	EN_US	Die Sprache der Dokumente.
MODELCCSID	CCSID der Datenbank	Die CCSID der Dokumentmodelldateien.
UPDATECOMMITCOUNT	0	Die Anzahl der Änderungen, die bei einer Aktualisierung in einer Transaktion verarbeitet werden.
CLEARCOMMITCOUNT	0	Die Anzahl der Änderungen, die während der Ausführung eines Befehls CLEAR INDEX in einer Transaktion verarbeitet werden.
UPDATEFREQUENCY	NONE	Zeitplan für die Überprüfung auf Aktualisierungen in neuen Indizes.
UPDATEMINIMUM	1	Die Mindestanzahl von Änderungen, bevor die Aktualisierung ausgeführt wird.
WORKDIRECTORY	Siehe Pfadname unter 'Ammerkungen'.	Verzeichnis für temporäre Indexdateien. Bei Linux- und UNIX-Betriebssystemen: <i>INSTHOME</i> /sql1lib/db2ext/indexes. Dabei ist <i>INSTHOME</i> das Ausgangsverzeichnis des Instanzeigners. Bei Windows-Betriebssystemen: <i>INSTPROFDIR</i> \instanzname\db2ext\indexes. Dabei ist <i>INSTPROFDIR</i> das Instanzprofilverzeichnis.

Tabelle 3. Sicht db2ext.dbdefaults (Forts.)

Attribut	Standardwert	Anmerkungen
CACHEDIRECTORY	Siehe Pfadname unter 'Anmerkungen'.	Standardverzeichnis für die Option PERSISTENT CACHE des Befehls CREATE INDEX . Bei Linux- und UNIX-Betriebssystemen: <i>INSTHOME</i> /sql11ib/db2ext/indexes. Dabei ist <i>INSTHOME</i> das Ausgangsverzeichnis des Instanzeigners. Bei Windows-Betriebssystemen: <i>INSTPROFDIR</i> \instanzname\db2ext\indexes. Dabei ist <i>INSTPROFDIR</i> das Instanzprofilverzeichnis.
PCTFREE	50	Der Prozentsatz des Cache, der für zukünftige Einfügungen freigehalten wird.
USERPERSISTENTCACHE	1	Persistenten Cache verwenden.
AUTOMATICREORG	1	Die Option REORGANIZE im Befehl CREATE INDEX . Dieser Wert legt eine automatische Reorganisation fest.
TREATNUMBERSASWORDS	0	Zeichen- und Zahlenfolgen werden auch dann nicht als getrennte Wörter interpretiert, wenn sie neben Zeichen stehen. Zum Beispiel bedeutet der Standardwert 0, dass 'tea42at5' als ein Wort betrachtet wird.
INDEXSTOPWORDS	1	Sämtliche Textelemente einschließlich Stoppwörtern indexieren.
VERSION		NSE V9.7 ist die aktuelle Versionsnummer von Net Search Extender.
UPDATEDELAY	0	Gibt die Zeitdauer in Sekunden an, die für Teilaktualisierungen ohne Erfassungstabellen benötigt wird. Nur Einträge, die älter als diese Zeitdauer sind, werden aus der Protokolltabelle entnommen. Dies dient zur Vermeidung verlorener Aktualisierungen. Ein Beispiel wären Dokumentänderungen, die in Transaktionsszenarios, in denen Benutzertransaktionen zu Problemen mit Aktualisierungsbefehlen führen, nicht im Index wiedergegeben werden. Aus diesem Grund sollte für den Parameter UpdateDelay die maximal zulässige Zeitdauer einer Benutzerschreibtransaktion für die Tabelle angegeben werden, für die der Index erstellt wurde.

Tabelle 3. Sicht *db2ext.dbdefaults* (Forts.)

Attribut	Standardwert	Anmerkungen
AUXLOGNORM	OFF	Aktivieren Sie die erweiterte textverwaltete Zwischenspeicherinfrastruktur für nichtpartitionierte Tabellen nicht standardmäßig. Die Zwischenspeicherinfrastruktur kann für einen Textindex mit der expliziten Indexkonfiguration AUXLOG ON aktiviert werden.
AUXLOGPART	ON	Aktivieren Sie standardmäßig die erweiterte textverwaltete Zwischenspeicherinfrastruktur für bereichspartitionierte Tabellen. Die Zwischenspeicherinfrastruktur kann für einen Textindex mit der expliziten Indexkonfiguration AUXLOG OFF inaktiviert werden.
LOCKSCHEULERFILE	0	Legen Sie den Wert '1' fest, um gleichzeitigen Schreibzugriff auf die Schedulerdatei <i>ctedem.dat</i> zu blockieren. In einigen Szenarien, die einen hohen Grad der Parallelität bei Erstellungen, Löschungen oder bei Änderungen der Aktualisierungshäufigkeit aufweisen, werden Aktualisierungsprozesse sonst unnötig gestartet.

Anmerkung: Unter Windows-Betriebssystemen wurde das Standardindexverzeichnis geändert. In DB2 Net Search Extender Version 9.5 lautete das Verzeichnis *db2-installationspfad\db2ext\indexes*. Informationen zum DB2-Installationspfad finden Sie in Verzeichnisstruktur für das installierte DB2-Datenbankprodukt (Windows).

Sichten für Informationen auf Indexebene

Informationen auf Indexebene können mithilfe von Sichten abgerufen werden.

Aus den folgenden Net Search Extender-Sichten können Sie Informationen auf Indexebene abrufen:

- `db2ext.textindexes`
- `db2ext.textindexformats`
- `db2ext.indexconfiguration`
- `<schema für namen der indexereignissicht>.<name der indexereignissicht>`

Aus Gründen der Abwärtskompatibilität werden die Sichten `db2ext.textcolumns`, `db2ext.formats` und `db2ext.models` von DB2 Text Information Extender zwar weiterhin unterstützt, ihre Verwendung wird jedoch nicht mehr empfohlen.

Beachten Sie, dass die Spalten `OPERATION`, `OPERATIONBEGIN` und `OPERATIONEND` in der Sicht `db2ext.textcolumns` nicht mehr unterstützt werden.

Sicht 'db2ext.textindexes'

Jede Datenbank, die für Net Search Extender aktiviert ist, enthält eine Sicht db2ext.textindexes. Diese Sicht enthält Informationen zu Einstellungen, Statistiken und Standardwerten für die erstellten Textindizes in dieser Datenbank.

Wenn Sie einen Textindex erstellen, werden die folgenden neuen Einträge in db2ext.textindexes erstellt. Wenn Sie die Textindizes löschen, werden diese Einträge ebenfalls gelöscht.

Sie können die Sicht abfragen, um Informationen über die Indizes zu erhalten. Das folgende Beispiel zeigt eine Abfrage über das Indexschema:

```
db2 "select COLNAME from DB2EXT.TEXTINDEXES where INDSHEMA='meinschema'
    and INDNAME='meinindex'"
```

Beachten Sie jedoch, dass Sie die Sicht weder mithilfe normaler SQL-Befehle zur Datenbearbeitung modifizieren noch die Katalogsicht explizit erstellen oder löschen können. Weitere Inhalte der Sicht finden Sie in der folgenden Tabelle.

Beachten Sie auch, dass die Replikationsparameter nicht in dieser Sicht enthalten sind.

Tabelle 4. Sicht 'db2ext.textindexes'

Attribut	Typ	Kommentare
INDSHEMA	VARCHAR(128)	Schemaname des Textindex.
INDNAME	VARCHAR(128)	Name des Textindex.
TABSHEMA	VARCHAR(128)	Der Tabellename des Schemas für Basistabellen, Kurznamen und Sichten.
TABNAME	VARCHAR(128)	Der Aliasname, für den der Index erstellt wurde.
COLNAME	VARCHAR(128)	Spalte, für die der Index erstellt wurde.
CCSID	INTEGER	Die CCSID der Dokumente für diesen Index.
LANGUAGE	VARCHAR(5)	Die Dokumentsprache für diesen Index.
FUNCTIONSCHEMA	VARCHAR(128)	Schema der Spaltenzuordnungsfunktion.
FUNCTIONNAME	VARCHAR(18)	Name der Spaltenzuordnungsfunktion.
INDEXDIRECTORY	VARCHAR(256)	Verzeichnis für Volltextindexdateien.
WORKDIRECTORY	VARCHAR(256)	Verzeichnis für temporäre Indexdateien.
CACHEDIRECTORY	VARCHAR(256)	Verzeichnis für den persistenten Cache (wenn persistentcache=1).
UPDATEFREQUENCY	VARCHAR(300)	Auslösekriterium für die Ausführung automatischer Aktualisierungen dieses Index.
UPDATEMINIMUM	INTEGER	Mindestanzahl von Dokumenten, die geändert werden müssen, bevor eine Aktualisierung durchgeführt wird.
EVENTVIEWSHEMA	VARCHAR(128)	Schema der Ereignissicht, die für diesen Index erstellt wurde.
EVENTVIEWNAME	VARCHAR(128)	Name der Ereignissicht, die für diesen Index erstellt wurde.

Tabelle 4. Sicht 'db2ext.textindexes' (Forts.)

Attribut	Typ	Kommentare
LOGVIEWSCHEMA	VARCHAR(128)	Schema der Protokollsicht, die für diesen Index erstellt wurde.
LOGVIEWNAME	VARCHAR(128)	Name der Protokollsicht, die für einen Index erstellt wurde (wichtig für Teilaktualisierungen an Sichten).
COMMITCOUNT	INTEGER	Standardwert für Anzahl der Commits bei der Aktualisierung.
NUMBER_DOCS	INTEGER	Die Gesamtanzahl von Dokumenten, die sich zurzeit im Index befinden. Hierbei ist zu beachten, dass während einer Indexaktualisierung dieser Wert nur aktualisiert wird, wenn commitcount gesetzt wurde.
REORG_SUGGESTED	INTEGER	Gibt an, ob die Leistung durch Ausführen eines Befehls UPDATE INDEX REORGANIZE verbessert werden kann. Dieser Parameter ist nur wahr (1), wenn für mindestens einen der Knoten eine Indexreorganisation vorgeschlagen wurde.
REORGAUTOMATIC	INTEGER	1, wenn der Index während der Aktualisierungsoperation automatisch reorganisiert wird.
RECREATEONUPDATE	INTEGER	1, wenn der Index während der Aktualisierungsoperation automatisch reorganisiert wird.
CREATIONTIME	TIMESTAMP	Zeitpunkt der Indexerstellung.
UPDATETIME	TIMESTAMP	Zeitpunkt der letzten Aktualisierung. Wenn die Werte für UPDATE TIME und CREATION TIME identisch sind, wurde keine Aktualisierung durchgeführt.
PERSISTENTCACHE	INTEGER	1, wenn ein persistenter Cache verwendet wird.
MAXIMUMCACHESIZE	INTEGER	Maximale Größe des Caches.
PCTFREE	INTEGER	Prozentsatz, zu dem der Cache für zukünftige Einfügungen freigehalten wird.
CACHETABLE	VARCHAR(32000)	Spaltenausdrucksliste für die im Cache gespeicherte Tabelle (CACHE TABLE).
RESULTORDER	VARCHAR(32000)	SQL-Klausel ORDER BY für INITIAL RESULT ORDER.
ATTRIBUTES	VARCHAR(32000)	Spaltenausdrucksliste für ATTRIBUTES.
VIEWKEYCOLUMNS	VARCHAR(32000)	Schlüsselspalten für den Index für die Sicht.
AUXSTAGINGSHEMA	VARCHAR(16)	Schema für textverwaltete Zwischenspeichertabellen; auf SYSIBMTS festgelegt

Tabelle 4. Sicht 'db2ext.textindexes' (Forts.)

Attribut	Typ	Kommentare
AUXSTAGINGNAME	VARCHAR(48)	Vom System generierter Name der textverwalteten Zwischenspeichertabelle (nur falls konfiguriert).

Sicht 'db2ext.indexconfiguration'

Die Sicht 'db2ext.indexconfiguration' enthält Parameter der Indexkonfiguration.

Auf diese Sicht kann über normale SQL-Abfragefunktionen zugegriffen werden. Das folgende Beispiel zeigt eine Abfrage über den Indexnamen:

```
db2 "select VALUE from DB2EXT.INDEXCONFIGURATION where INDSHEMA='meinschema'
and INDNAME='meinindex' and PARAMETER ='INDEXSTOPWORDS'"
```

Weitere Inhalte der Sicht finden Sie in den folgenden Tabellen.

Tabelle 5. Sicht 'db2ext.indexconfiguration'

Attribut	Typ	Kommentare
INDSHEMA	VARCHAR(128)	Schemaname des Index.
INDNAME	VARCHAR(128)	Name des Index.
PARAMETER	VARCHAR(30)	Typ des Parameters.
VALUE	VARCHAR(512)	Wert des Parameters.

Für die Attribute PARAMETER und VALUE stehen mehrere Werte zur Verfügung.

Tabelle 6. Sicht 'db2ext.indexconfiguration'

Attribut und Werte	Attribut und Werte
PARAMETER	VALUE
- TREATNUMBERASWORDS	- 0 oder 1
- INDEXSTOPWORDS	- 0 oder 1
- UPDATEDELAY	- Sekunden >= 0
AUXLOGPART	- ON oder OFF
AUXLOGNORM	- ON oder OFF

Weitere Informationen finden Sie in der Beschreibung der Option **CONFIGURATION** des Befehls **CREATE INDEX**.

Sicht 'db2ext.textindexformats'

Die Sicht 'db2ext.textindexformats' enthält Format- und Modellinformationen für Indizes.

Das folgende Beispiel zeigt eine Abfrage über den Indexnamen:

```
db2 "select FORMAT from DB2EXT.TEXTINDEXFORMATS where INDSHEMA='meinschema'
and INDNAME='meinindex'"
```

Weitere Inhalte der Sicht finden Sie in der folgenden Tabelle.

Tabelle 7. Sicht 'db2ext.textindexformats'

Attribut	Typ	Anmerkungen
INDSCHEMA	VARCHAR(128)	Schemaname für den Index (dient als Präfix für den Indexnamen und den Schemanamen in der Protokolltabelle).
INDNAME	VARCHAR(128)	Der im Befehl CREATE INDEX angegebene Indexname.
FORMAT	VARCHAR(30)	Das Modell ist an dieses Format gebunden.
MODELNAME	VARCHAR(30)	Der Name eines Dokumentmodells.
MODELFILE	VARCHAR(256)	Die Datei, in der die Modelldefinition enthalten ist.
MODELCCSID	INTEGER	CCSID der MODELFILE.
DEFAULT	INTEGER	Zurzeit 1, da mehrere Formate in einem Index gegenwärtig nicht unterstützt werden.

Tabellensichten für einen Textindex

Mithilfe der Ereignissicht und der Protokolltabellensicht können Sie Informationen auf Indexebene abfragen.

Aus den folgenden DB2 Net Search Extender-Sichten können Sie Informationen auf Indexebene abrufen:

- Ereignissicht
- Protokolltabellensicht

Ereignissicht

Diese Sicht gibt Ihnen die Möglichkeit, Informationen zum Indexierungsstatus und zu Fehlerereignissen abzufragen, wenn bei der Indexierung Probleme auftreten, wenn also z. B. ein Dokument nicht gefunden werden kann. Solche Ereignisse der Indexaktualisierung werden in die Ereignistabelle des Index geschrieben.

Diese Sicht gibt Ihnen die Möglichkeit, Informationen zum Indexierungsstatus und zu Fehlerereignissen abzufragen, wenn bei der Indexierung Probleme auftreten, wenn also z. B. ein Dokument nicht gefunden werden kann. Solche Ereignisse der Indexaktualisierung werden in die Ereignistabelle des Index geschrieben.

Das Schema und der Name werden in der Sicht 'db2ext.textindexes' gespeichert. Den Namen der Ereignissicht können Sie wie im folgenden Beispiel ermitteln:

```
db2 "select EVENTVIEWSHEMA, EVENTVIEWNAME from DB2EXT.TEXTINDEXES
      where INDSCHEMA = 'meinschema' and INDNAME = 'meinindex'"
```

Die Ereignissicht für einen Index besteht aus folgenden Spalten.

Tabelle 8. Die Ereignissicht

Attribut	Typ	Kommentare
OPERATION	INTEGER	Die für die Benutzertabelle durchgeführte Operation, die im Volltextindex nachvollzogen wird (INSERT = 0/ UPDATE = 1/ DELETE = 2). Wenn Sie eine Tabelle der Capture-Komponente bei der Datenreplikation verwenden, werden Aktualisierungsoperationen in eine Lösch- und eine Einfügeoperation aufgeteilt. In diesem Fall kann eine Einfügeoperation in der Ereignistabelle entweder von einer Einfüge- oder einer Aktualisierungsoperation aus in der Tabelle, in der der Index erstellt wurde, vorgenommen werden.
TIME	TIMESTAMP	Zeitmarke für die Erstellung des Ereigniseintrags.
REASON	INTEGER	Der Ursachencode. Eine Liste der Ursachencodes finden Sie in Kapitel 64, „Ursachencodes der Textsuchsteuerkomponente“, auf Seite 297.
SEVERITY	INTEGER	Die Wertigkeit des Tabelleneintrags. Hierbei steht 1 beispielsweise für eine Information, 4 für eine Warnung und 8 für einen Tabelleneintragsfehler.
MESSAGE	VARCHAR(1024)	Zusätzliche Textinformationen.
KEY1, ... KEY14	Abhängig von der Benutzertabelle	Erste Primärschlüsselspalte der Benutzertabelle bis zur letzten Primärschlüsselspalte (maximal 14).
PARTITION	INTEGER	Die Datenbankpartitionsnummer, bei der dieser Fehler aufgetreten ist. In einer Umgebung mit nicht partitionierten Datenbanken lautet der Wert 0.

Die Ereignisse können mithilfe des Befehls **DB2TEXT CLEAR EVENTS** gelöscht werden.

Anmerkung: Informative Ereignisse wie das Starten, das Commit und das Beenden einer Aktualisierungsverarbeitung sind in dieser Sicht ebenfalls verfügbar.

In diesem Fall enthalten die Spalten Key1, ... Key14 und OPERATION alle Nullwerte.

Bei Indizes auf der Basis von Sichten beziehen sich die Spalten PK01, ..., PK14 auf die Spalten, die in der Klausel KEY COLUMNS des Befehls **CREATE INDEX** angegeben wurden.

Protokolltabellen, Sichten und Kurznamen

Der Zweck der Protokolltabelle besteht darin, die Änderungsoperationen an der Benutzertabelle oder Sicht zu speichern, die eine Synchronisation mit dem externen Volltextindex erfordern.

Für Indizes, die für reguläre Tabellen oder Kurznamentabellen erstellt werden, werden Trigger für die Benutzertabelle erstellt, die die Änderungsinformationen in

die Protokolltabelle einfügen. Wenn jedoch Replikationserfassungstabellen verwendet werden, wird keine Protokolltabelle erstellt. Stattdessen wird die Replikationserfassungstabelle verwendet.

Bei Protokolltabellen liest der UPDATE-Befehl die Einträge und löscht diese, nachdem die Synchronisation erfolgreich ausgeführt wurde.

Im Fall von Indizes auf der Basis von Sichten kann die Protokolltabelle jedoch nicht mithilfe von Triggern gefüllt werden. Da die Sicht aktualisiert werden kann, trägt der Benutzer die Verantwortung für diese Aufgabe.

Tabelle 9. Die Protokolltabellensicht

Attribut	Typ	Kommentare
OPERATION	INTEGER	Der Typ der Änderung, die an der Benutzertabelle vorgenommen wurde und eine Indexsynchronisation erfordert (0 = INSERT, 1 = UPDATE, 2 = DELETE).
TIME	TIMESTAMP	Die Zeitmarke für die Erstellung einer Zeile in dieser Tabelle.
PK01 ... PKnm	Wie bei Benutzertabelle	Bei einem Fehler die Spalte, in der der Fehler aufgetreten ist. Es handelt sich um Kopien der Primärschlüsselspalten der Benutzertabelle oder, bei einem Index auf der Basis einer Sicht, der entsprechenden Schlüsselspalten.

Der Benutzer, der die Tabelle erstellt, kann für diese Sicht eine SELECT-, UPDATE-, INSERT- oder DELETE-Operation ausführen.

Tabelle 10. Die textverwaltete Zwischenspeichertabelle

Attribut	Typ	Anmerkungen
PK 1..n	Wie bei Benutzertabelle	Eine Kopie der primären Schlüsseldefinition(en) der Benutzertabelle
Globaltransid	CHAR(8)	Interne Transaktions-ID
Globaltranstime	CHAR(13)	Zeitmarke
Operationtype	Integer	1 einfügen
		-1 löschen

Der Inhalt der Tabelle wird von der Textsuche verwaltet, aber ein Administrator kann Einträge aus der Tabelle löschen.

Wenn Sie im Befehl CREATE INDEX eine Replikationserfassungstabelle angeben, wird keine Protokolldatei erstellt und statt dieser wird die Replikationserfassungstabelle verwendet. Die Replikationserfassungstabelle muss die folgenden Spalten umfassen:

Tabelle 11. Die Replikationserfassungstabelle

Attribut	Typ	Anmerkungen
IBMSNAP_OPERATION	INTEGER	Der Typ der Änderung, die an der CD-Tabelle (CD = Change Data) oder CCD-Tabelle (CCD = Consistent Change Data) vorgenommen wurde und eine Indexsynchronisation erfordert (I = INSERT, U = UPDATE, D = DELETE). Wenn Sie eine Tabelle der Capture-Komponente bei der Datenreplikation verwenden, werden Aktualisierungsoperationen in eine Lösch- und eine Einfügeoperation aufgeteilt. In diesem Fall kann eine Einfügeoperation in der Ereignistabelle entweder von einer Einfüge- oder einer Aktualisierungsoperation aus in der Tabelle, in der der Index erstellt wurde, vorgenommen werden.
IBMSNAP_COMMITSEQ	CHAR	Wird der entsprechenden Spalte der CD- bzw. CCD-Tabelle zugeordnet.
IBMSNAP_INTENTSEQ	CHAR	Wird der entsprechenden Spalte der CD- bzw. CCD-Tabelle zugeordnet.
PK01 ... PKnm	Wie bei Benutzertabelle	Bei einem Fehler die Spalte, in der der Fehler aufgetreten ist. Dabei handelt es sich um die Primärschlüsselspalten der Benutzertabelle.

Der Benutzer, der die Tabelle definiert, erhält die Berechtigung zur Durchführung von SELECT-, UPDATE-, INSERT- und DELETE-Operationen.

Kapitel 64. Ursachencodes der Textsuchsteuerkomponente

Die Textsuchsteuerkomponente kann Fehlercodes und die zugehörigen Fehlerursachen ausgeben.

- 0 Operation wurde erfolgreich ausgeführt - kein Fehler aufgetreten.
- 1 Einer Funktion wurde eine ungültige Kennung übergeben.
- 2 Funktion konnte nicht ausreichend Speicher zuordnen.
- 3 Funktion konnte wegen Zugriffsbeschränkungen oder Sicherheitseinschränkungen nicht ausgeführt werden.
- 4 Die Operation wird für diese Laufzeitversion der Textsuchsteuerkomponente nicht unterstützt.
- 5 Die Operation ist zurzeit nicht aktiviert.
- 6 Die Anwendung hat das Protokoll der Textsuchsteuerkomponente durch Aufrufen der Funktionen der Textsuchsteuerkomponente in regelwidriger Reihenfolge verletzt.
- 7 Ein unerwarteter Fehler ist aufgetreten. Melden Sie diesen Fehler Ihrem Serviceansprechpartner.
- 8 Eine ungültige Sprache wurde angegeben.
- 9 Die angegebene Sprache ist gültig, wird jedoch von der Laufzeitumgebung der Textsuchsteuerkomponente nicht unterstützt.
- 10 Eine ungültige ID für codierten Zeichensatz (CCSID) wurde angegeben.
- 11 Die angegebene ID für codierten Zeichensatz (CCSID) ist gültig, wird jedoch von der Laufzeitumgebung der Textsuchsteuerkomponente nicht unterstützt.
- 12 Eine ungültige Dokument-ID wurde angegeben.
- 13 Das angegebene Dokumentformat ist gültig, wird jedoch von der Laufzeitumgebung der Textsuchsteuerkomponente nicht unterstützt.
- 14 Ein ungültiges Dokumentformat wurde angegeben.
- 15 Die Operation konnte wegen einer Zugriffsbeschränkung bei der Datei-Eingabe/Ausgabe nicht erfolgreich durchgeführt werden.
- 16 Die Operation konnte wegen Lesefehlern bei der Datei-Eingabe/Ausgabe nicht erfolgreich durchgeführt werden.
- 17 Die Operation konnte wegen Lesefehlern bei der Datei-Eingabe nicht erfolgreich durchgeführt werden.
- 18 Die Operation konnte wegen Schreibfehlern bei der Dateiausgabe nicht erfolgreich durchgeführt werden.
- 19 Die Operation konnte wegen Suchfehlern bei der Datei-Eingabe/Ausgabe nicht erfolgreich durchgeführt werden.
- 20 Die Operation konnte wegen Tell-Fehlern bei der Datei-Eingabe/Ausgabe nicht erfolgreich durchgeführt werden.
- 21 Die Operation konnte wegen Schließfehlern bei der Datei-Eingabe/Ausgabe nicht erfolgreich durchgeführt werden.

- 22 Die Operation konnte wegen Fehlern bei Umbenennungsoperationen nicht erfolgreich durchgeführt werden.
- 23 Die Operation konnte wegen Fehlern bei Löschoptionen nicht erfolgreich durchgeführt werden.
- 24 Die Operation konnte wegen Fehlern bei Verzechniserstellungsoperationen nicht erfolgreich durchgeführt werden.
- 25 Mindestens ein Funktionsargument hatte einen ungültigen Wert (z. B. einen unerwarteten Nullzeiger oder einen ungültigen Aufzählungstypwert).
- 26 Das angegebene Verzeichnis ist nicht vorhanden.
- 27 In der Textsuchsteuerkomponente ist ein unerwarteter Fehler aufgetreten. Untersuchen Sie den Fehlercode der Textsuchsteuerkomponente im Fehlerinformationsobjekt auf weitere Details.
- 28 Es ist ein unerwarteter COS-Fehler aufgetreten. Bitte melden Sie diesen Fehler.
- 29 Es wurde versucht, ein leeres Dokument zu aktualisieren.
- 30 Das angegebene Argument wird für diese Operation nicht unterstützt.
- 31 Der Parser für das Datumsattribut hat bei dem Versuch, ein Datumsattribut zu analysieren, einen ungültigen Wert gefunden.
- 32 Der Parser für numerische Attribute (NUMBER) hat bei dem Versuch, ein numerisches Attribut zu analysieren, einen ungültigen Wert gefunden.
- 33 Attributname ungültig, wahrscheinlich zu lang.
- 35 Reservierte Nummer für zukünftige Verwendung.
- 36 Das Eingabedokument enthält ein Attribut (DATE, NUMBER oder STRING), das die Längenbegrenzung für Attribute überschreitet. Der Attributtext wurde auf diese Begrenzung abgeschnitten.
- 38 Der vom Benutzer festgelegte Warnungsschwellenwert wurde überschritten. Infolgedessen wurde dieser Fehler generiert.
- 39 Das Eingabedokument konnte nicht indexiert werden. Es enthält zu viele verschachtelte Felder.
- 40 Die Begrenzung für verschiedene Attribute für einen der Attributtypen wurde für diesen Index überschritten.
- 46 Der Iterator ist nicht (bzw. nicht mehr) gültig, da seine Liste leer ist oder gelöscht wurde.
- 47 Die Funktion wird für die übergebene Art von Kennung (handle) nicht unterstützt. Dieser Fehler tritt auf, wenn zum Beispiel versucht wird, die Funktion `itlQueryResultEntryObtainData` auf einen Listeniterator anzuwenden, der keinen Abfrageergebnisiterator darstellt.
- 48 Diese Warnung wird ausgegeben, wenn eine Stoppwörterdatei für die angegebene Sprache und den Ressourcenpfad nicht gefunden werden kann.
- 49 Diese Warnung wird ausgegeben, wenn eine Stoppwörterdatei keine Stoppwörter enthält.
- 50 Diese Warnung wird ausgegeben, wenn eine Stoppwörterdatei ungültige Daten enthält.
- 100 Der Index konnte nicht geöffnet werden, weil er nicht mit dem angegebenen Namen und im angegebenen Verzeichnis vorhanden ist.

- 101 Der angegebene Indexname ist kein gültiger Indexname.
- 102 Das angegebene Indexverzeichnis ist kein gültiges Indexverzeichnis.
- 103 Die Operation kann nicht ausgeführt werden, weil die Textsuchsteuerkomponente eine Beschädigung der Indexstruktur oder der Indexdateigruppen erkannt hat.
- 104 Der angegebene Index kann nicht erstellt werden, weil er bereits mit dem angegebenen Namen und in dem angegebenen Verzeichnis vorhanden ist.
- 109 Bevor eine andere Operation für diesen Index ausgeführt werden kann, muss ein Rollback ausgeführt werden.
- 110 Die Indexkonfigurationsdatei enthält den im Fehlerkontext angegebenen verbindlichen Abschnitt nicht.
- 111 Die Indexkonfigurationsdatei enthält die im Fehlerkontext angegebene verbindliche Option nicht.
- 112 Die Indexkonfigurationsdatei enthält ungültige Daten in der im Fehlerkontext angegebenen Option.
- 113 Die Indexkonfigurationsdatei entspricht nicht der Version der Textsuchsteuerkomponente.
- 200 Der angegebene Dokumentmodellname ist kein gültiger Modellname.
- 201 Der angegebene Name eines Dokumentmodellfelds ist kein gültiger Feldname.
- 202 Das angegebene Dokumentmodell ist nicht bekannt.
- 203 Das angegebene Dokumentmodell ist bereits vorhanden und kann nicht erneut definiert werden.
- 204 Dem Index wurden zu viele oder zu große Dokumentmodelle hinzugefügt.
- 205 Das Dokumentmodell enthält zu viele Elemente.
- 206 Das Dokumentmodellelement enthält einen Parameter (XML-Attribut), der für diesen Elementtyp nicht zulässig ist.
- 207 Das Dokumentmodellelement enthält einen Parameterwert, der für diesen Parametertyp (XML-Attribut) nicht zulässig ist.
- 208 Das Dokumentmodellelement enthält einen erforderlichen Parameter (XML-Attribut) nicht, zum Beispiel 'name'.
- 209 Das Dokumentmodell scheint kein XML-Format zu haben oder beginnt mit einem unerwarteten XML-Element.
- 210 Der angegebene XPath-Wert (Querverweiswert) enthält ein unerwartetes Token.
- 211 Der angegebene XPath-Wert (Querverweiswert) enthält eine unerwartete Achsenkennung (d. h., einen Namen mit zwei folgenden Doppelpunkten).
- 212 Der angegebene XPath-Wert (Querverweiswert) enthält einen unerwarteten Knotentest (NodeTest).
- 213 Die Verzeichnisdatei für das Dokumentmodell (Erweiterung .mdx) ist beschädigt.
- 214 Die Indexdatei für das Dokumentmodell (Erweiterung .mox) ist beschädigt.

- 215 Das Dokument enthält ein XML-Element, das einem Dokumentattribut zugeordnet ist und das ein weiteres Dokumentattribut enthält. Das innere Attribut wird ignoriert.
- 216 Der angegebene Parameterwert ist als GPP- oder HTML-Tag zu lang.
- 217 Das Dokumentmodell enthält eine doppelte Felddefinition.
- 218 Das Dokumentmodell enthält eine doppelte Attributdefinition.
- 300 Die Operation kann nicht ausgeführt werden, weil die Textsuchsteuerkomponente eine Beschädigung in den Indexdateien erkannt hat, die zur Zuordnung von Dokumentnamen verwendet werden.
- 301 Die Operation kann nicht ausgeführt werden, weil die Textsuchsteuerkomponente eine ungültige Dokumentnummer erkannt hat.
- 302 Die Operation kann nicht ausgeführt werden, weil die Textsuchsteuerkomponente eine ungültige Dokumentkennung erkannt hat.
- 303 Die Operation kann nicht ausgeführt werden, weil die Textsuchsteuerkomponente keinen Indexeintrag für die Dokumentkennung gefunden hat.
- 304 Die Operation kann nicht ausgeführt werden, weil die Textsuchsteuerkomponente keinen Indexeintrag für die Dokumentnummer gefunden hat.
- 305 Die Operation kann nicht ausgeführt werden, weil die Textsuchsteuerkomponente einen Überlauf in den verwendeten Dokumentnummern festgestellt hat.
- 306 Die Dokumentkennung, die die Anwendung zu indexieren versucht hat, ist bereits in der Liste von Dokumenten aufgetreten. Die Textsuchsteuerkomponente unterstützt keine doppelten Dokumentkennungen, die in einer Indexierungssequenz auftreten, d. h. bevor die Aktualisierung festgeschrieben wurde.
- 340 Die Begriffsstärke ist nicht gültig.
- 341 Die Relationsnummer ist nicht gültig.
- 342 Der Relationstyp ist nicht gültig. Verwenden Sie einen der DEFINE-Werte, die in der API beschrieben sind.
- 343 Die Phrase (Begriff) ist zu lang.
- 344 Unerwartetes Dateiende beim Lesen angetroffen.
- 345 Beim Lesen der Index-/Thesaurusdateien wurde ein Versionskonflikt festgestellt.
- 346 Überlauf in Thesauruspuffer.
- 347 Ungültiger Name für eine Datei oder ein Verzeichnis. Name wahrscheinlich zu lang.
- 348 Suche hat einen Begriff (Phrase) im Wörterverzeichnis nicht gefunden, oder ein Eintrag in der Definitionsdatei enthält einen verbindlichen Begriff nicht.
- 349 Definitionsdatei ist leer.
- 350 Thesauruswörterverzeichnis oder Definitionsdatei ist nicht wie über Eingabeparameter angegeben vorhanden.
- 351 Syntaxfehler in Definitionsdatei.
- 352 Die Relation wurde falsch angegeben.
- 352 Die Relationsnummer lag außerhalb des Bereichs.

- 360 Ein ungültiges Platzhalterzeichen für Einzelzeichen wurde verwendet.
- 361 Ein ungültiges Platzhalterzeichen für mehrere Zeichen wurde verwendet.
- 362 Die Operatorstelligkeit ist kleiner als die Anzahl der in der Abfrage angegebenen Operanden.
- 363 Operatorwert außerhalb des durch die ItEnOperator-Aufzählung definierten Bereichs.
- 364 Der Wert für die Rangformel liegt außerhalb des Aufzählungsbereichs.
- 365 Die Nummer zur Angabe des Umgebungssegments liegt außerhalb des Bereichs.
- 366 Die Abfrage ist im Aufbau und kann nicht umdefiniert oder zurückgesetzt werden.
- 367 Der Bereich (Scope), der als voriges Suchergebnis angegeben wurde, bezeichnet ein leeres Ergebnis.
- 368 Ungültiger Aufruf, der das Hinzufügen von Feldnamen vor der Einstellung des ersten Feldes anfordert.
- 369 Eine ungültige Suchmarkierung (Flag), die einen ungültigen Vergleich mit dem Indexinhalt anfordert, wird ignoriert. Wenn zum Beispiel ein Vergleich mit Beachtung der Groß-/Kleinschreibung für einen Index angefordert wurde, der ohne Unterscheidung der Groß-/Kleinschreibung erstellt wurde, wird dieser Ursachencode in den Fehlerinformationen gezeigt.
- 370 Die Verwendung von Platzhalterzeichen in Zeichenfolgen wird für Thailändisch oder DBCS-Sprachen nicht unterstützt.
- 371 Keine gültige Abfrageeingabe. Beispielsweise sind die Suchbegriffe verfügbar.
- 372 Ungültige Vergleichsoperationen angefordert.
- 373 Ungültige Vergleichsoperationen angefordert.
- 374 Die Suchindexkennung wurde für einen leeren Index angefordert.
- 375 Die Kombination von Operator und angefordertem Operatormodus wird nicht unterstützt.
- 380 Das Suchergebnis ist nicht vollständig. Die Suche wurde wegen eines Schwellenwerts abgebrochen.
- 381 Die Indexsuche hat gezeigt, dass die Abfrage Stoppwörter enthielt.
- 401 Die Operation kann nicht ausgeführt werden, weil die Textsuchsteuerkomponente eine Beschädigung in den Indexdateien erkannt hat, die zur Zuordnung von Feld-/Attributnamen verwendet werden.
- 402 Die Operation kann nicht ausgeführt werden, weil die Textsuchsteuerkomponente einen ungültigen Feld- oder Attributnamen erkannt hat.
- 403 Die Operation kann nicht ausgeführt werden, weil der angegebene Feld- oder Attributname unbekannt ist.
- 404 Die Begrenzung für verschiedene Attribute für einen der Attributtypen oder verschiedener Felder wurde für diesen Index überschritten.
- 500 Das Dokument bzw. die Daten enthalten eine ungültige Zeichenfolge (in einer UTF8-, UTF16- oder DBCS-Quelle).
- 501 Die Codepageumwandlung war fehlerhaft.

- 502 Das Dokument bzw. die Daten enthalten eine unvollständige Zeichenfolge (in einer UTF8-, UTF16- oder DBCS-Quelle).
- 503 Der Codepage-Converter enthält einen ungültigen Deskriptor.
- 600 Das XML-Dokument enthält eine asynchrone Entität, zum Beispiel einen XML-Attributwert ohne Anführungszeichen.
- 602 Ungültiger Zeichenverweis (zum Beispiel 'oder').
- 603 Ungültiger Verweis für binäre Entität.
- 604 XML-Parser Expat konnte nicht erstellt werden.
- 605 Ein Attributname in einem Tag muss eindeutig sein.
- 607 XML-Parser hat einen ungültigen Verweis auf externe Entität gefunden.
- 608 Ein Dokument enthält ein falsches Token, zum Beispiel ein fehlendes < oder >.
- 609 XML-Dokumente müssen einen umschließenden Tag haben, und nach diesem umschließenden Endtag ist nur ein Tag und kein Text zulässig.
- 610 Eine Verarbeitungsanweisung ist an ihrer Position nicht zulässig. Zum Beispiel ist die erste Verarbeitungsanweisung nicht der Prolog <?xml ..?>.
- 611 Ein Element ist eine Folge aus Starttag, Inhalt und Endtag. Dieser Fehler ist zum Beispiel in einer Sequenz '<s> text /s>' aufgetreten, weil der Endtag nicht korrekt ist.
- 612 Die Speicherzuordnung im XML-Parser ist fehlgeschlagen.
- 614 Ungültiger Parameterentitätsverweis.
- 615 Ein unvollständiges Zeichen, vielleicht nur das erste Byte eines zwei Byte langen UTF8-Zeichens.
- 616 Rekursiver Entitätsverweis.
- 617 XML-Syntaxfehler. Zum Beispiel Text außerhalb der umschließenden Start- und Endtags.
- 618 Jeder Starttag benötigt einen entsprechenden Endtag.
- 619 Nicht geschlossener CDATA-Abschnitt.
- 620 Nicht geschlossenes Token. Zum Beispiel Text nach dem letzten Token in einem Dokument.
- 621 Das Dokument enthält eine Entität, die nicht aufgelöst werden konnte.
- 622 Unerwarteter Fehler.
- 631 Feld- oder Attributinformationen im Metatag konnten nicht syntaktisch analysiert werden. Der Tag muss das Format <meta name="abc" content="xyz"> aufweisen. Möglicherweise sind die Attribute 'name' oder 'content' des Metatags nicht korrekt.
- 632 Die Entität konnte nicht in ein Zeichen umgewandelt werden.
- 650 Verschiedene Felddefinitionen beginnen mit dem gleichen Starttag.
- 651 Ein Starttag enthält einen weiteren, sodass die Tags nicht eindeutig sind.
- 652 Wenn ein Feld und ein Attribut den gleichen Starttag verwenden, müssen sie den gleichen Endtag oder beide keinen Endtag verwenden.
- 653 Ein Feld ist noch nicht geschlossen, wenn das Dokument endet.

- 654 Für das strukturierte Format ist kein Dokumentmodell angegeben. Das Dokument wird als einfaches Textdokument ohne Feld- oder Attributinformationen analysiert.
- 670 Die Operation konnte nicht ausgeführt werden, weil sie die 'Outside In'-Bibliotheken (TM) erfordert, die nicht gefunden werden konnten.
- 671 Die Operation konnte nicht ausgeführt werden, weil eine erforderliche Prozedur aus den 'Outside In' (TM)-Bibliotheken nicht geladen werden konnte. Wahrscheinlich sind die Bibliotheken veraltet oder beschädigt.
- 672 Es ist ein Fehler aufgetreten, als das Dokument mit 'Outside In' verarbeitet wurde.

Teil 14. Fehlerbehebung

Kapitel 65. Tracefunktion zur Fehleranalyse

Wenn Sie einem IBM Ansprechpartner einen Fehler melden müssen, werden Sie möglicherweise gebeten, die Tracefunktion zu aktivieren, sodass Informationen in eine Datei geschrieben werden, die zur Lokalisierung des Fehlers verwendet werden kann.

Informationen zu diesem Vorgang

Da die Systemleistung bei aktivierter Tracefunktion beeinträchtigt wird, verwenden Sie die Tracefunktion nur nach den Anweisungen eines Ansprechpartners des IBM Support Center bzw. eines Mitarbeiters Ihrer technischen Unterstützung.

Vorgehensweise

- Wenn Sie die Tracefunktion aktivieren möchten, verwenden Sie die DB2-Funktion:
`db2trc on`
- Wenn Sie bestimmte Informationen zu Net Search Extender benötigen, können Sie eine Platzhalterzeichenfolge mit der Komponente 96 verwenden:
`db2trc on -m *.*.96.*.*`

Nächste Schritte

Bei schwerwiegenden Fehlern kann es darüber hinaus nützlich sein, die Protokoll-datei **db2diag** zu prüfen.

Kapitel 66. DB2-Objekte ohne die richtigen Net Search Extender-Befehle löschen

Tabelle löschen

Bevor Sie eine Tabelle mit einem oder mehreren Indizes löschen, müssen Sie einen bestimmten Befehl eingeben.

Informationen zu diesem Vorgang

Bevor Sie eine Tabelle mit einem oder mehreren Indizes löschen, müssen Sie für jeden Textindex den folgenden Befehl eingeben:

```
db2text drop index indexname for text
```

Wenn Sie vor dem Löschen der Indizes versehentlich eine Tabelle löschen, sind dennoch Indexteile vorhanden, wie z. B. die Verwaltungstabellen und Textindexdateien.

Löschen Sie die Indizes mit dem Befehl **db2text drop index**, um diese Dateien zu entfernen, auch wenn die Tabelle nicht mehr vorhanden ist.

Datenbank löschen

Bevor Sie eine Datenbank mit einem oder mehreren Indizes löschen, müssen Sie für jeden Textindex einen entsprechenden Befehl eingeben. Wenn Sie diesen Befehl nicht verwenden, müssen Sie alle Indexdateien im Indexverzeichnis und im Indexarbeitsverzeichnis manuell löschen.

Vorgehensweise

Bevor Sie eine Datenbank mit einem oder mehreren Textindizes löschen, müssen Sie Folgendes ausführen:

1. Setzen Sie für jeden Textindex den folgenden Befehl ab:

```
db2text drop index indexname for text
```

Wenn Sie diesen Befehl nicht verwenden, müssen Sie alle Indexdateien im Indexverzeichnis und im Indexarbeitsverzeichnis manuell löschen.

2. Sie müssen die Schedulerdatei `ctedem.dat` bearbeiten, wenn die Indizes, die zur gelöschten Datenbank gehören, während einer automatischen Aktualisierung erstellt wurden.
 - a. Geben Sie Folgendes ein:
 - Für UNIX:

```
db2text stop force  
cd ~/sql1lib/db2text
```
 - Für Windows:

```
db2text stop force  
cd db2-installationspfad\sql1lib\name_der_db2-instanz\db2text
```
 - b. Öffnen Sie im Verzeichnis die Datei `ctedem.dat`, und entfernen Sie alle Einträge, die auf die gelöschte Datenbank verweisen.

Kapitel 67. Installationsrückkehrcodes unter Windows

Der folgende Abschnitt enthält Informationen zu Installationsrückkehrcodes unter Windows.

setup.exe-Rückkehrcodes unter Windows

Das Programm `setup.exe` gibt die folgenden Codes in der Datei `setup.log` zurück:

- 0 Erfolg
- -1 Allgemeiner Fehler
- -2 Ungültiger Modus
- -3 Erforderliche Daten in der Datei `setup.iss` nicht gefunden
- -4 Nicht ausreichend Speicher verfügbar
- -5 Datei nicht vorhanden
- -6 In Antwortdatei kann nicht geschrieben werden
- -7 In Protokolldatei kann nicht geschrieben werden
- -8 Ungültiger Pfad zur im Hintergrund ausgeführten Antwortdatei zur Installationsabschirmung (`.iss`)
- -9 Kein gültiger Listentyp
- -10 Ungültiger Datentyp
- -11 Unbekannter Fehler bei der Einrichtung
- -12 Dialogfelder funktionieren nicht
- -51 Der angegebene Ordner kann nicht erstellt werden
- -52 Auf die angegebene Datei oder den angegebenen Ordner kann nicht zugegriffen werden
- -53 Ungültige Option ausgewählt

Kapitel 68. Hinweise und Tipps

Dieser Abschnitt enthält nützliche Hinweise und Tipps zu Themen wie zum Beispiel Berechtigung, Sprache, Ändern von `db2cli.ini`-Dateien und Client-Server-Interoperabilität.

Berechtigung

Wenn Sie den Befehl **DB2TEXT START** unter Windows absetzen, stellen Sie sicher, dass Sie zur Gruppe der Administratoren gehören. Ansonsten schlägt der Befehl **DB2TEXT START** fehl, und es wird die folgende Nachricht zurückgegeben: CTE0218 Funktion 'OpenSCManager()' ist fehlgeschlagen, Fehlercode '5'.

Berechtigung

Stellen Sie unter Windows sicher, dass der Net Search Extender-Instanzservice `DB2EXT-DB2_instanzname` mit einem Benutzerkonto und nicht mit dem Systemkonto ausgeführt wird. Wenn Sie das Systemkonto verwenden, können Sie Ihre Datenbank nicht aktivieren.

Sprache

Da Net Search Extender-Ereignisprotokollnachrichten immer in der Sprache des DB2-Servers ausgegeben werden, werden Ereignisprotokollnachrichten für Befehle, die über den DB2-Befehlszeilenprozessor abgesetzt werden, möglicherweise in einer anderen Sprache angezeigt.

Modifizieren der Datei `db2cli.ini`

Wenn Sie die Datei `db2cli.ini` geändert haben und Probleme mit Net Search Extender haben, stellen Sie die Originalversion der Datei `db2cli.ini` wieder her.

Interoperabilität des Client-Servers

Sie müssen sowohl auf dem Server als auch auf dem Client dieselbe Net Search Extender-Fixpackversion installieren.

Die Interoperabilität zwischen Client und Server und umgekehrt ist nur bei unterstützten Plattform möglich.

Protokollgröße

Wenn der Indexierungsprozess nicht ausgeführt wird, da die Fehlermeldungen und Warnungen mehr Speicherplatz erfordern, als für die DB2-Protokollgröße verfügbar ist, führt DB2 für die gesamte Transaktion einen Rollback durch und schreibt die Protokolltabelleneinträge nicht fest. Dies bedeutet, dass Sie die Einträge nicht anzeigen können.

In der DB2-Dokumentation finden Sie Informationen dazu, wie Sie den Wert der Größe Ihres Transaktionsprotokolls erhöhen können, damit es nicht zu dieser Situation kommt.

DBCS-Objektnamen

Wenn Sie DBCS-Objektnamen in **db2text**-Verwaltungsbefehlen verwenden, müssen Sie diese Namen in Anführungszeichen setzen, damit keine Umsetzung in Großbuchstaben erfolgt.

Indexteilaktualisierung für Kurznamen

Wenn Anfangsaktualisierungen für zwei oder mehr Indizes gleichzeitig gestartet werden, gibt der Aktualisierungsbefehl möglicherweise den Fehler `SQL0803N` zurück. Wiederholen Sie in diesem Fall den Aktualisierungsbefehl.

Einzelplatzhalterzeichen und Zeichennormalisierung

Wörter wie 'über' werden normalisiert und im Index in ihrer normalisierten Form ('ueber') gespeichert. Deshalb können Sie 'über' nicht finden, wenn Sie eine Abfrage absetzen, die Einzelplatzhalterzeichen wie z. B. '_ber' enthält.

Doppelte Cachespaltennamen verwenden

Wenn Sie doppelte Cachespaltennamen verwenden, wird bei der Textindexerstellung oder der Indexaktualisierung zwar kein Fehler ausgegeben, aber Sie können keinen Suchvorgang durchführen. Wenn Sie versuchen, eine Suche durchzuführen, wird eine SQL-Fehlernachricht ausgegeben, die darauf hinweist, dass doppelte Spalten verwendet wurden.

Falsche Größe des gemeinsam genutzten Speichers

Wenn die angegebene maximale Cachegröße in dem Befehl **db2text activate cache** zu klein ist, ist die erforderliche Cachegröße, die in der resultierenden Fehlernachricht angezeigt wird, falsch.

Mit den Funktionen DB2EXT.MAXIMUM_CACHE_SIZE und DB2EXT.PCTFREE können Sie die richtige Cachegröße ermitteln. Korrigieren Sie die maximale Cachegröße mithilfe des Befehls **db2text alter index**, und aktivieren Sie den Cache noch einmal.

Unicodetabellen in einer anderen Datenbank (nicht Unicode)

Sie dürfen keinen Textindex für eine Unicodetabelle erstellen, wenn Ihre Datenbank Unicode nicht unterstützt.

Falsche Codepage für die Variable LANG in einer Abfrage unter Linux

Bei Verwendung einer 7-Bit-ASCII-Codepage für die Variable **LANG** in einer Abfrage wird der folgende Fehler angezeigt: Bei der Verarbeitung des Sucharguments ist ein Fehler aufgetreten.

Damit es dazu nicht kommt, ändern Sie die Variable **LANG** in einen 8-Bit-Wert **LANG**, starten Sie DB2 neu, und wiederholen Sie den Suchvorgang.

Dateizugriffsprobleme

Wenn Sie keinen Zugriff auf Dateien haben (z. B. Dokumentmodell, Textindex oder Thesaurus), müssen Sie sicherstellen, dass Sie das richtige Kennwort verwendet haben und über die korrekte Berechtigung zum Ausführen der Net Search Extender-Instanzservices verfügen. Dies gilt besonders für gemeinsam genutzte Ressourcen in zugeordneten Netzlaufwerken.

Cache kann nicht verwendet werden

Während einer Suche oder Aktivierung eines Caches wird möglicherweise die folgende Fehlernachricht angezeigt: CTE0271 Cache nicht verwendbar, DEACTIVATE und ACTIVATE RECREATE erforderlich. Überprüfen Sie Ihre Systemeinstellungen, und vergrößern Sie die Paging-Rate und den freien Speicher, um dieses Problem zu beheben.

Instanzservices nach Deinstallation nicht gelöscht

Wenn die Instanzservices nach der Deinstallation von Net Search Extender nicht gelöscht werden, können Sie das folgende Tool zum manuellen Löschen der Services verwenden: `ctereg instanzname unregister`. Beispiel: `ctereg db2-0 unregister`.

Unter UNIX wird 'ctepcrx' abnormal beendet.

Prüfen Sie, ob der verwendete Instanzeigner über eine separate abgeschirmte Benutzer-ID verfügt. Öffnen Sie hierzu die Datei mit der Endung `.fenced` im Verzeichnis `instanzausgangsverz_/sql11ib/adm`, und prüfen Sie, ob der Instanzeigner zugleich der abgeschirmte Benutzer ist.

Die Indexaktualisierung schlägt mit SQL0668N fehl.

Wenn die Indexaktualisierung mit SQL0668N (rc=1) fehlschlägt, war der Zugriff auf die textverwaltete Zwischenspeichertabelle blockiert, da eine Datenbankoperation für die Basistabelle durchgeführt wurde, die Integrationsverarbeitung für abhängige Tabellen erfordert.

Verwenden Sie den Befehl **RESET PENDING**, um die Blockierung der Zwischenspeichertabelle aufzuheben, und führen Sie den Befehl zur Indexaktualisierung erneut aus.

Wenn skalare SQL-Suchfunktionen nicht für Tabellen verwendet werden können, die sich über mehrere Partitionen erstrecken

In Situationen, in denen die Textsuchfunktion nicht direkt auf die partitionierte Tabelle verweist oder wenn sie ein Mitglied eines Subselects ist, der die Klausel OUTER JOIN enthält, gibt die Suche einen Fehler zurück (SQL0270N rc=109). Falls dies auftritt, versuchen Sie, die Abfrage manuell neu und in einem anderen Format zu schreiben und sie erneut auszugeben.

Fehler CTE0249 beim Überprüfen des Setups der Umgebung mit partitionierten Datenbanken.

Führen Sie die folgenden Aktionen aus, wenn Sie unter Windows das Setup der Umgebung mit partitionierten Datenbanken überprüfen und der Fehler CTE0249 Das ausführbare Programm 'ctepisc' wurde abnormal beendet auftritt:

1. Überprüfen Sie für jeden Computer in der Konfiguration der Umgebung mit partitionierten Datenbanken, ob die Umgebung für partitionierte DB2-Server vorbereitet wurde, entsprechend den Anweisungen in . Stellen Sie insbesondere sicher, dass in der **Active Directory Users and Computers**-Steuerkonsole bei jedem Computerkonto im Dialogfeld **Eigenschaften** auf der Registerkarte **Allgemein** das Kontrollkästchen zum Akzeptieren eines Computers für Delegation ausgewählt ist.
2. Stellen Sie sicher, dass der Windows-Dienst 'DB2 Remote Command Server' auf allen teilnehmenden Computern gestartet wurde.

Fehler CTE0150E beim Erstellen eines Textindex für eine bereichspartitionierte Tabelle ohne die Angabe der Klausel ADMINISTRATION TABLES IN.

Wenn Sie versuchen, einen Textindex für eine bereichspartitionierte Tabelle zu erstellen, müssen Sie stets die Klausel ADMINISTRATION TABLES IN angeben. Weitere Informationen hierzu finden Sie im Abschnitt CTE0150E.

Teil 15. Anhänge und Schlussteil

Anhang A. Übersicht über technische Informationen zu DB2

Technische Informationen zu DB2 liegen in verschiedenen Formaten vor, die auf unterschiedliche Weise abgerufen werden können.

Die technischen Informationen zu DB2 stehen über die folgenden Tools und Methoden zur Verfügung:

- DB2 Information Center
 - Themen (zu Tasks, Konzepten und Referenzinformationen)
 - Beispielprogramme
 - Lernprogramme
- DB2-Bücher
 - PDF-Dateien (für den Download verfügbar)
 - PDF-Dateien (auf der DB2-PDF-DVD)
 - Gedruckte Bücher
- Hilfe für Befehlszeile
 - Hilfe für Befehle
 - Hilfe für Nachrichten

Anmerkung: Die Themen des DB2 Information Center werden häufiger aktualisiert als die PDF- und Hardcopybücher. Um stets die neuesten Informationen zur Verfügung zu haben, sollten Sie die Dokumentationsaktualisierungen installieren, sobald diese verfügbar sind, oder das DB2 Information Center unter ibm.com aufrufen.

Darüber hinaus können Sie auf zusätzliche technische Informationen zu DB2, wie beispielsweise technische Hinweise (Technotes), White Papers und IBM Redbooks, online über ibm.com zugreifen. Rufen Sie dazu die Website 'DB2 Information Management - Software - Library' unter <http://www.ibm.com/software/data/sw-library/> auf.

Feedback zur Dokumentation

Senden Sie uns Ihr Feedback zur DB2-Dokumentation! Wenn Sie Anregungen zur Verbesserung der DB2-Dokumentation haben, senden Sie eine E-Mail an db2docs@ca.ibm.com. Das DB2-Dokumentationsteam bearbeitet das gesamte Feedback, kann jedoch nicht im Einzelnen auf Ihre E-Mails antworten. Nennen Sie uns, wenn möglich, konkrete Beispiele, sodass wir die Problemstellung besser beurteilen können. Wenn Sie uns Feedback zu einem bestimmten Thema oder einer bestimmten Hilfedatei senden, geben Sie den entsprechenden Titel sowie die URL an.

Verwenden Sie diese E-Mail-Adresse nicht, wenn Sie sich an den DB2-Kundendienst wenden möchten. Wenn ein technisches Problem bei DB2 vorliegt, das Sie mithilfe der Dokumentation nicht beheben können, fordern Sie beim zuständigen IBM Service-Center Unterstützung an.

Bibliothek mit technischen Informationen zu DB2 im Hardcopy- oder PDF-Format

Die folgenden Tabellen enthalten eine Beschreibung der DB2-Bibliothek, die im IBM Publications Center unter www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss zur Verfügung steht. Über die folgende Adresse können Sie englische Handbücher im PDF-Format sowie übersetzte Versionen zu DB2 Version 10.1 herunterladen: www.ibm.com/support/docview.wss?rs=71&uid=swg27009474.

In den Tabellen sind die Bücher, die in gedruckter Form zur Verfügung stehen, gekennzeichnet; möglicherweise sind diese in Ihrem Land oder Ihrer Region jedoch nicht verfügbar.

Die Formnummer wird bei jeder Aktualisierung eines Handbuchs erhöht. Anhand der nachfolgenden Liste können Sie sicherstellen, dass Sie die jeweils neueste Version des Handbuchs lesen.

Anmerkung: Das *DB2 Information Center* wird häufiger aktualisiert als die PDF- und Hardcopybücher.

Tabelle 12. Technische Informationen zu DB2

Name	IBM Form	In gedruckter Form verfügbar	Letzte Aktualisierung
<i>Administrative API Reference</i>	SC27-3864-00	Ja	April 2012
<i>Administrative Routines and Views</i>	SC27-3865-01	Nein	Januar 2013
<i>Call Level Interface Guide and Reference Volume 1</i>	SC27-3866-01	Ja	Januar 2013
<i>Call Level Interface Guide and Reference Volume 2</i>	SC27-3867-01	Ja	Januar 2013
<i>Command Reference</i>	SC27-3868-01	Ja	Januar 2013
<i>Datenbankverwaltung - Konzepte und Konfiguration - Referenzinformationen</i>	SC12-4673-01	Ja	Januar 2013
<i>Dienstprogramme für das Versetzen von Daten - Handbuch und Referenz</i>	SC12-4691-01	Ja	Januar 2013
<i>Datenbanküberwachung - Handbuch und Referenz</i>	SC12-4674-01	Ja	Januar 2013
<i>Datenrecovery und hohe Verfügbarkeit - Handbuch und Referenz</i>	SC12-4692-01	Ja	Januar 2013
<i>Datenbanksicherheit</i>	SC12-4693-01	Ja	Januar 2013
<i>DB2 Workload Management - Handbuch und Referenz</i>	SC12-4683-01	Ja	Januar 2013
<i>Developing ADO.NET and OLE DB Applications</i>	SC27-3873-01	Ja	Januar 2013

Tabelle 12. Technische Informationen zu DB2 (Forts.)

Name	IBM Form	In gedruckter Form verfügbar	Letzte Aktualisierung
<i>Developing Embedded SQL Applications</i>	SC27-3874-01	Ja	Januar 2013
<i>Developing Java Applications</i>	SC27-3875-01	Ja	Januar 2013
<i>Developing Perl, PHP, Python, and Ruby on Rails Applications</i>	SC27-3876-00	Nein	April 2012
<i>Developing RDF Applications for IBM Data Servers</i>	SC27-4462-00	Ja	Januar 2013
<i>Developing User-defined Routines (SQL and External)</i>	SC27-3877-01	Ja	Januar 2013
<i>Getting Started with Database Application Development</i>	GI13-2046-01	Ja	Januar 2013
<i>Installation und Verwaltung von DB2 unter Linux und Windows - Erste Schritte</i>	GI11-3285-00	Ja	April 2012
<i>Globalisierung</i>	SC12-4694-00	Ja	April 2012
<i>DB2-Server - Installation</i>	SC12-4677-01	Ja	Januar 2013
<i>IBM Data Server-Clients - Installation</i>	SC12-4678-00	Nein	April 2012
<i>Fehlernachrichten, Band 1</i>	SC12-4686-01	Nein	Januar 2013
<i>Fehlernachrichten, Band 2</i>	SC12-4687-01	Nein	Januar 2013
<i>Net Search Extender - Verwaltung und Benutzerhandbuch</i>	SC12-4689-01	Nein	Januar 2013
<i>Partitionierung und Clustering</i>	SC12-4695-01	Ja	Januar 2013
<i>Preparation Guide for DB2 10.1 Fundamentals Exam 610</i>	SC27-4540-00	Nein	Januar 2013
<i>Preparation Guide for DB2 10.1 DBA for Linux, UNIX, and Windows Exam 611</i>	SC27-4541-00	Nein	Januar 2013
<i>pureXML - Handbuch</i>	SC12-4684-01	Ja	Januar 2013
<i>Spatial Extender - Benutzer- und Referenzhandbuch</i>	SC12-4688-00	Nein	April 2012
<i>SQL Procedural Languages: Application Enablement and Support</i>	SC27-3896-01	Ja	Januar 2013
<i>SQL Reference Volume 1</i>	SC27-3885-01	Ja	Januar 2013

Tabelle 12. Technische Informationen zu DB2 (Forts.)

Name	IBM Form	In gedruckter Form verfügbar	Letzte Aktualisierung
SQL Reference Volume 2	SC27-3886-01	Ja	Januar 2013
Text Search	SC12-4690-01	Ja	Januar 2013
Fehlerbehebung und Optimieren der Datenbankleistung	SC12-4675-01	Ja	Januar 2013
Upgrade auf DB2 Version 10.1	SC12-4676-01	Ja	Januar 2013
Neuerungen in DB2 Version 10.1	SC12-4682-01	Ja	Januar 2013
XQuery - Referenz	SC12-4685-01	Nein	Januar 2013

Tabelle 13. Technische Informationen zu DB2 Connect

Name	IBM Form	In gedruckter Form verfügbar	Letzte Aktualisierung
DB2 Connect - Installation und Konfiguration von DB2 Connect Personal Edition	SC12-4679-00	Ja	April 2012
DB2 Connect - Installation und Konfiguration von DB2 Connect-Servern	SC12-4680-01	Ja	Januar 2013
DB2 Connect - Benutzerhandbuch	SC12-4681-01	Ja	Januar 2013

Aufrufen der Hilfe für den SQL-Status über den Befehlszeilenprozessor

DB2-Produkte geben für Bedingungen, die aufgrund einer SQL-Anweisung generiert werden können, einen SQLSTATE-Wert zurück. Die SQLSTATE-Hilfe erläutert die Bedeutung der SQL-Statuswerte und der SQL-Statusklassencodes.

Vorgehensweise

Zum Starten der Hilfe für SQL-Statuswerte müssen Sie den Befehlszeilenprozessor öffnen und Folgendes eingeben:

? SQL-Status oder ? Klassencode

Hierbei steht *SQL-Status* für einen gültigen fünfstelligen SQL-Statuswert und *Klassencode* für die ersten beiden Ziffern dieses Statuswerts.

So kann beispielsweise durch die Eingabe von ? 08003 Hilfe für den SQL-Statuswert 08003 angezeigt werden, durch die Eingabe von ? 08 Hilfe für den Klassencode 08.

Zugriff auf verschiedene Versionen des DB2 Information Center

Die Dokumentation für andere Versionen der DB2-Produkte finden Sie in den jeweiligen Information Centers unter ibm.com.

Informationen zu diesem Vorgang

Für Themen aus DB2 Version 10.1 lautet die URL für das *DB2 Information Center* <http://publib.boulder.ibm.com/infocenter/db2luw/v10r1>.

Für Themen aus DB2 Version 9.8 lautet die URL des *DB2 Information Center* <http://publib.boulder.ibm.com/infocenter/db2luw/v9r8/>.

Für Themen aus DB2 Version 9.7 lautet die URL des *DB2 Information Center* <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/>.

Für Themen aus DB2 Version 9.5 lautet die URL des *DB2 Information Center* <http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/>.

Für Themen aus DB2 Version 9.1 lautet die URL des *DB2 Information Center* <http://publib.boulder.ibm.com/infocenter/db2luw/v9/>.

Für Themen aus DB2 Version 8 lautet die URL des *DB2 Information Center* <http://publib.boulder.ibm.com/infocenter/db2luw/v8/>.

Aktualisieren des auf Ihrem Computer oder Intranet-Server installierten DB2 Information Center

Ein lokal installiertes DB2 Information Center muss regelmäßig aktualisiert werden.

Vorbereitende Schritte

Ein DB2 Version 10.1 Information Center muss bereits installiert sein. Einzelheiten hierzu finden Sie unter „Installation des DB2 Information Center mit dem DB2-Installationsassistenten“ in *DB2-Server - Installation*. Alle für die Installation des Information Center geltenden Voraussetzungen und Einschränkungen gelten auch für die Aktualisierung des Information Center.

Informationen zu diesem Vorgang

Ein vorhandenes DB2 Information Center kann automatisch oder manuell aktualisiert werden:

- Mit automatischen Aktualisierungen werden vorhandene Komponenten und Sprachen des Information Center aktualisiert. Ein Vorteil von automatischen Aktualisierungen ist, dass das Information Center im Vergleich zu einer manuellen Aktualisierung nur für einen kurzen Zeitraum nicht verfügbar ist. Darüber hinaus können automatische Aktualisierungen so konfiguriert werden, dass sie als Teil anderer, regelmäßig ausgeführter Stapeljobs ausgeführt werden.
- Mit manuellen Aktualisierungen können Sie vorhandene Komponenten und Sprachen des Information Center aktualisieren. Automatische Aktualisierungen reduzieren die Ausfallzeiten während des Aktualisierungsprozesses, Sie müssen jedoch den manuellen Prozess verwenden, wenn Sie Komponenten oder Sprachen hinzufügen möchten. Beispiel: Ein lokales Information Center wurde ursprünglich sowohl mit englischer als auch mit französischer Sprachunterstützung installiert; nun soll auch die deutsche Sprachunterstützung installiert werden. Bei einer manuellen Aktualisierung werden sowohl eine Installation der deutschen Sprachunterstützung als auch eine Aktualisierung der vorhandenen Komponenten und Sprachen des Information Center durchgeführt. Sie müssen jedoch bei einer manuellen Aktualisierung das Information Center manuell stop-

pen, aktualisieren und erneut starten. Das Information Center ist während des gesamten Aktualisierungsprozesses nicht verfügbar. Während des automatischen Aktualisierungsprozesses kommt es zu einem Ausfall des Information Center, und es wird erst wieder nach der Aktualisierung erneut gestartet.

Dieser Abschnitt enthält Details zum Prozess der automatischen Aktualisierung. Anweisungen zur manuellen Aktualisierung finden Sie im Abschnitt „Manuelles Aktualisieren des auf Ihrem Computer oder Intranet-Server installierten DB2 Information Center“.

Vorgehensweise

Gehen Sie wie folgt vor, um das auf Ihrem Computer bzw. Intranet-Server installierte DB2 Information Center automatisch zu aktualisieren:

1. Unter Linux:
 - a. Navigieren Sie zu dem Pfad, in dem das Information Center installiert ist. Standardmäßig ist das DB2 Information Center im Verzeichnis `/opt/ibm/db2ic/V10.1` installiert.
 - b. Navigieren Sie vom Installationsverzeichnis in das Verzeichnis `doc/bin`.
 - c. Führen Sie das Script `update-ic` aus:
`update-ic`
2. Unter Windows:
 - a. Öffnen Sie ein Befehlsfenster.
 - b. Navigieren Sie zu dem Pfad, in dem das Information Center installiert ist. Standardmäßig ist das DB2 Information Center im Verzeichnis `<Programme>\IBM\DB2 Information Center\Version 10.1` installiert, wobei `<Programme>` das Verzeichnis der Programmdateien angibt.
 - c. Navigieren Sie vom Installationsverzeichnis in das Verzeichnis `doc\bin`.
 - d. Führen Sie die Datei `update-ic.bat` aus:
`update-ic.bat`

Ergebnisse

Das DB2 Information Center wird automatisch erneut gestartet. Standen Aktualisierungen zur Verfügung, zeigt das Information Center die neuen und aktualisierten Abschnitte an. Waren keine Aktualisierungen für das Information Center verfügbar, wird eine entsprechende Nachricht zum Protokoll hinzugefügt. Die Protokolldatei befindet sich im Verzeichnis `doc\eclipse\configuration`. Der Name der Protokolldatei ist eine Zufallszahl. Beispiel: `1239053440785.log`.

Manuelles Aktualisieren des auf Ihrem Computer oder Intranet-Server installierten DB2 Information Center

Wenn Sie das DB2 Information Center lokal installiert haben, können Sie Dokumentationsaktualisierungen von IBM abrufen und installieren.

Informationen zu diesem Vorgang

Zur manuellen Aktualisierung des lokal installierten *DB2 Information Center* sind die folgenden Schritte erforderlich:

1. Stoppen Sie das *DB2 Information Center* auf Ihrem Computer und starten Sie das Information Center im Standalone-Modus erneut. Die Ausführung des Information Center im Standalone-Modus verhindert, dass andere Benutzer in Ihrem

Netz auf das Information Center zugreifen, und ermöglicht das Anwenden von Aktualisierungen. Die Workstationversion des DB2 Information Center wird stets im Standalone-Modus ausgeführt.

2. Verwenden Sie die Aktualisierungsfunktion, um zu prüfen, welche Aktualisierungen verfügbar sind. Falls Aktualisierungen verfügbar sind, die Sie installieren müssen, können Sie die Aktualisierungsfunktion verwenden, um diese abzurufen und zu installieren.

Anmerkung: Wenn es in der verwendeten Umgebung erforderlich ist, die Aktualisierungen für das *DB2 Information Center* auf einer Maschine zu installieren, die nicht über eine Verbindung zum Internet verfügt, spiegeln Sie die Aktualisierungssite auf ein lokales Dateisystem und verwenden Sie dabei eine Maschine, die mit dem Internet verbunden ist und auf der das *DB2 Information Center* installiert ist. Wenn viele Benutzer Ihres Netzes die Dokumentationsaktualisierungen installieren sollen, können Sie die Zeit, die jeder einzelne Benutzer für die Aktualisierungen benötigt, reduzieren, indem Sie die Aktualisierungssite lokal spiegeln und ein Proxy dafür erstellen.

Ist dies der Fall, verwenden Sie die Aktualisierungsfunktion, um die Pakete abzurufen. Die Aktualisierungsfunktion ist jedoch nur im Standalone-Modus verfügbar.

3. Stoppen Sie das im Standalone-Modus gestartete Information Center und starten Sie das *DB2 Information Center* auf Ihrem Computer erneut.

Anmerkung: Unter Windows 2008 und Windows Vista (und neueren Versionen) müssen die in diesem Abschnitt aufgeführten Befehle mit Administratorberechtigung ausgeführt werden. Zum Öffnen einer Eingabeaufforderung oder eines Grafiktools mit vollen Administratorberechtigungen klicken Sie mit der rechten Maustaste die Verknüpfung an und wählen Sie **Als Administrator ausführen** aus.

Vorgehensweise

Gehen Sie wie folgt vor, um das auf Ihrem Computer bzw. Intranet-Server installierte *DB2 Information Center* zu aktualisieren:

1. Stoppen Sie das *DB2 Information Center*.
 - Unter Windows: Klicken Sie **Start > Systemsteuerung > Verwaltung > Dienste** an. Klicken Sie mit der rechten Maustaste das **DB2 Information Center** an und wählen Sie **Beenden** aus.
 - Unter Linux: Geben Sie den folgenden Befehl ein:

```
/etc/init.d/db2icdv10 stop
```
2. Starten Sie das Information Center im Standalone-Modus.
 - Unter Windows:
 - a. Öffnen Sie ein Befehlsfenster.
 - b. Navigieren Sie zu dem Pfad, in dem das Information Center installiert ist. Standardmäßig ist das *DB2 Information Center* im Verzeichnis `Programme\IBM\DB2 Information Center\Version 10.1` installiert, wobei `Programme` das Verzeichnis der Programmdateien angibt.
 - c. Navigieren Sie vom Installationsverzeichnis in das Verzeichnis `doc\bin`.
 - d. Führen Sie die Datei `help_start.bat` aus:

```
help_start.bat
```
 - Unter Linux:

- a. Navigieren Sie zu dem Pfad, in dem das Information Center installiert ist. Standardmäßig ist das *DB2 Information Center* im Verzeichnis `/opt/ibm/db2ic/V10.1` installiert.
- b. Navigieren Sie vom Installationsverzeichnis in das Verzeichnis `doc/bin`.
- c. Führen Sie das Script `help_start` aus:

```
help_start
```

Der standardmäßig auf dem System verwendete Web-Browser wird geöffnet und zeigt die Standalone-Version des Information Center an.

3. Klicken Sie die Aktualisierungsschaltfläche (🔄) an. (JavaScript muss im verwendeten Browser aktiviert sein.) Klicken Sie im rechten Fenster des Information Center die Schaltfläche für die Suche nach Aktualisierungen an. Eine Liste der Aktualisierungen für die vorhandene Dokumentation wird angezeigt.
4. Wählen Sie zum Initiieren des Installationsprozesses die gewünschten Aktualisierungen aus und klicken Sie anschließend die Schaltfläche für die Installation der Aktualisierungen an.
5. Klicken Sie nach Abschluss des Installationsprozesses **Fertigstellen** an.
6. Stoppen Sie das im Standalone-Modus gestartete Information Center:
 - Unter Windows: Navigieren Sie innerhalb des Installationsverzeichnisses zum Verzeichnis `doc\bin`, und führen Sie die Datei `help_end.bat` aus:

```
help_end.bat
```

Anmerkung: Die Stapeldatei `help_end` enthält die Befehle, die erforderlich sind, um die Prozesse, die mit der Stapeldatei `help_start` gestartet wurden, ordnungsgemäß zu stoppen. Verwenden Sie nicht die Tastenkombination `Strg+C` oder eine andere Methode, um `help_start.bat` zu stoppen.

- Unter Linux: Navigieren Sie innerhalb des Installationsverzeichnisses zum Verzeichnis `doc/bin`, und führen Sie das Script `help_end` aus:

```
help_end
```

Anmerkung: Das Script `help_end` enthält die Befehle, die erforderlich sind, um die Prozesse, die mit dem Script `help_start` gestartet wurden, ordnungsgemäß zu stoppen. Verwenden Sie keine andere Methode, um das Script `help_start` zu stoppen.

7. Starten Sie das *DB2 Information Center* erneut.
 - Unter Windows: Klicken Sie **Start > Systemsteuerung > Verwaltung > Dienste** an. Klicken Sie mit der rechten Maustaste das **DB2 Information Center** an und wählen Sie **Start** aus.
 - Unter Linux: Geben Sie den folgenden Befehl ein:

```
/etc/init.d/db2icdv10 start
```

Ergebnisse

Im aktualisierten *DB2 Information Center* werden die neuen und aktualisierten Themen angezeigt.

DB2-Lernprogramme

Die DB2-Lernprogramme unterstützen Sie dabei, sich mit den unterschiedlichen Aspekten der DB2-Produkte vertraut zu machen. Die Lerneinheiten bieten eine in einzelne Schritte unterteilte Anleitung.

Vorbereitungen

Die XHTML-Version des Lernprogramms kann über das Information Center unter <http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/> angezeigt werden.

In einigen der Lerneinheiten werden Beispieldaten und Codebeispiele verwendet. Informationen zu bestimmten Voraussetzungen für die Ausführung der Tasks finden Sie in der Beschreibung des Lernprogramms.

DB2-Lernprogramme

Klicken Sie zum Anzeigen des Lernprogramms den Titel an.

„pureXML“ in *pureXML - Handbuch*

Einrichten einer DB2-Datenbank, um XML-Daten zu speichern und Basisoperationen mit dem nativen XML-Datenspeicher auszuführen.

Informationen zur Fehlerbehebung in DB2

Es steht eine breite Palette verschiedener Informationen zur Fehlerbestimmung und Fehlerbehebung zur Verfügung, um Sie bei der Verwendung von DB2-Datenbankprodukten zu unterstützen.

DB2-Dokumentation

Informationen zur Fehlerbehebung stehen im Handbuch *Fehlerbehebung und Optimieren der Datenbankleistung* oder im Abschnitt mit grundlegenden Informationen zu Datenbanken im *DB2 Information Center* zur Verfügung, darunter:

- Informationen zum Eingrenzen und Aufdecken von Problemen mithilfe der Diagnosetools und -dienstprogramme von DB2.
- Lösungsvorschläge zu den am häufigsten auftretenden Problemen.
- Ratschläge zum Lösen anderer Probleme, die bei Verwendung der DB2-Datenbankprodukte auftreten können.

IBM Support Portal

Im IBM Support Portal finden Sie Informationen zu Problemen und den möglichen Ursachen und Fehlerbehebungsmaßnahmen. Die Website mit technischer Unterstützung enthält Links zu den neuesten DB2-Veröffentlichungen, technischen Hinweisen (TechNotes), APARs (Authorized Program Analysis Reports) und Fehlerkorrekturen, Fixpacks sowie weiteren Ressourcen. Sie können diese Wissensbasis nach möglichen Lösungen für aufgetretene Probleme durchsuchen.

Sie können auf das IBM Support Portal über die folgende Website zugreifen: http://www.ibm.com/support/entry/portal/Overview/Software/Information_Management/DB2_for_Linux,_UNIX_and_Windows.

Bedingungen

Die Berechtigungen zur Nutzung dieser Veröffentlichungen werden Ihnen auf der Basis der folgenden Bedingungen gewährt.

Anwendbarkeit: Diese Bedingungen gelten zusätzlich zu den Nutzungsbedingungen für die IBM Website.

Persönliche Nutzung: Sie dürfen diese Veröffentlichungen für Ihre persönliche, nicht kommerzielle Nutzung unter der Voraussetzung vervielfältigen, dass alle Eigentumsvermerke erhalten bleiben. Sie dürfen diese Veröffentlichungen oder Teile dieser Veröffentlichungen ohne ausdrückliche Genehmigung von IBM nicht weitergeben, anzeigen oder abgeleitete Werke davon erstellen.

Kommerzielle Nutzung: Sie dürfen diese Veröffentlichungen nur innerhalb Ihres Unternehmens und unter der Voraussetzung, dass alle Eigentumsvermerke erhalten bleiben, vervielfältigen, weitergeben und anzeigen. Sie dürfen diese Veröffentlichungen oder Teile dieser Veröffentlichungen ohne ausdrückliche Genehmigung von IBM außerhalb Ihres Unternehmens nicht vervielfältigen, weitergeben, anzeigen oder abgeleitete Werke davon erstellen.

Rechte: Abgesehen von den hier gewährten Berechtigungen erhalten Sie keine weiteren Berechtigungen, Lizenzen oder Rechte (veröffentlicht oder stillschweigend) in Bezug auf die Veröffentlichungen oder darin enthaltene Informationen, Daten, Software oder geistiges Eigentum.

IBM behält sich das Recht vor, die in diesem Dokument gewährten Berechtigungen nach eigenem Ermessen zurückzuziehen, wenn sich die Nutzung der Veröffentlichungen für IBM als nachteilig erweist oder wenn die obigen Nutzungsbestimmungen nicht genau befolgt werden.

Sie dürfen diese Informationen nur in Übereinstimmung mit allen anwendbaren Gesetzen und Vorschriften, einschließlich aller US-amerikanischen Exportgesetze und Verordnungen, herunterladen und exportieren.

IBM übernimmt keine Gewährleistung für den Inhalt dieser Informationen. Diese Veröffentlichungen werden auf der Grundlage des gegenwärtigen Zustands (auf "as-is"-Basis) und ohne eine ausdrückliche oder stillschweigende Gewährleistung für die Handelsüblichkeit, die Verwendungsfähigkeit oder die Freiheit der Rechte Dritter zur Verfügung gestellt.

IBM Marken: IBM, das IBM Logo und ibm.com sind Marken oder eingetragene Marken der International Business Machines Corporation. Weitere Produkt- oder Servicenamen können Marken von IBM oder anderen Herstellern sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite www.ibm.com/legal/copytrade.shtml.

Anhang B. Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. Die Informationen über Produkte anderer Hersteller als IBM basieren auf den zum Zeitpunkt der ersten Veröffentlichung dieses Dokuments verfügbaren Informationen und können geändert werden.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing
IBM Europe, Middle East & Africa
Tour Descartes
2, avenue Gambetta
92066 Paris La Defense
France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuauflage veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Canada Limited
U59/3600
3600 Steeles Avenue East
Markham, Ontario L3R 9Z7
CANADA

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 Dokument aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können davon abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Diese Veröffentlichung kann Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes enthalten. Sie sollen nur die Funktionen des Lizenzprogramms 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.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind und Programmier Techniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Beispielprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Die Beispielprogramme werden ohne Wartung (auf "as-is"-Basis) und ohne jegliche Gewährleistung zur Verfügung gestellt. IBM haftet nicht für Schäden, die durch Verwendung der Beispielprogramme entstehen.

Kopien oder Teile der Beispielprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (Name Ihrer Firma) (Jahr). Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corp. abgeleitet. © Copyright IBM Corp. *_Jahr/Jahre angeben_*. Alle Rechte vorbehalten.

Marken

IBM, das IBM Logo und ibm.com sind Marken oder eingetragene Marken der IBM Corporation in den USA und/oder anderen Ländern. Weitere Produkt- oder Servicennamen können Marken von oder anderen Herstellern sein. IBM oder anderen Herstellern sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite „Copyright and trademark information“ unter www.ibm.com/legal/copytrade.shtml.

Die folgenden Namen sind Marken oder eingetragene Marken anderer Unternehmen.

- Linux ist eine eingetragene Marke von Linus Torvalds in den USA und/oder anderen Ländern.
- Java und alle auf Java basierenden Marken und Logos sind Marken oder eingetragene Marken von Oracle und/oder ihren verbundenen Unternehmen.
- UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.
- Intel, das Intel-Logo, Intel Inside, Intel Inside logo, Celeron, Intel SpeedStep, Itanium und Pentium sind Marken oder eingetragene Marken der Intel Corporation oder deren Tochtergesellschaften in den USA und anderen Ländern.
- Microsoft, Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

Weitere Unternehmens-, Produkt- oder Servicennamen können Marken anderer Hersteller sein.

Index

A

ACTIVATE CACHE, Befehl 220
Aktualisierungen
 DB2 Information Center 323, 324
ALTER INDEX, Befehl 221

B

Bedingungen
 Veröffentlichungen 328
Befehle
 CONTROL 207
 START 209
 STOP 211
Bemerkungen 329
Benutzeraufgabenbereiche
 Datenbankadministratoren 75
 DB2-Instanzeigner 75
 Net Search Extender 75
 Texttabelleneigner 75

C

checknseindex, Befehl
 Details 123
CLEAR EVENTS, Befehl 225
CONTAINS, Funktion 267
CONTROL, Befehl 207
COPYRIGHT, Befehl 252
CREATE INDEX, Befehl 227
ctemigrdx, Tool 53

D

Datenbankpartitionen
 Net Search Extender 61
DB2 Information Center
 Aktualisierung 323, 324
 Versionen 323
DB2EXT.HIGHLIGHT, Funktion 272
DB2EXT.TEXTSEARCH, Befehl 277
DB2EXT.TEXTSEARCH, Funktion 269
DB2EXTHL, Befehl 217
DB2EXTTH, Befehl 246
db2nse_deinstall, Befehl 254
db2nse_install, Befehl 253
db2nsels, Befehl 254
db2text-Befehle 19
DEACTIVATE CACHE, Befehl 242
Deinstallation
 Net Search Extender
 Befehl db2nse_deinstall 254
 Übersicht 39
 UNIX 39
 Windows 39
DISABLE DATABASE, Befehl 215
Dokumentation
 gedruckt 320
 Nutzungsbedingungen 328

Dokumentation (*Forts.*)
 PDF-Dateien 320
 Übersicht 319
Dokumentformate
 Filtersoftware Outside In 73
Dokumentmodelle
 Dokumentattribute
 Begrenzungen 202
 Übersicht 185
 Dokumenttypdefinition 199
 Outside In-Filtertechnologie 196
 Referenzinformationen 199
 Standard 186
 Textfelder
 Begrenzungen 202
 Übersicht 185
 Übersicht 184
 unverschlüsselter Text
 Beispiel 184
 Definition 187
 XML 191
DROP INDEX (Net Search Extender-Befehl) 244

E

ENABLE DATABASE, Befehl 213
Extern gespeicherte Daten
 Übersicht 10

F

Fehlerbehebung
 Lernprogramme 327
 Onlineinformationen 327
Fehlerbestimmung
 Lernprogramme 327
 verfügbare Informationen 327
Fehlersuche 307
Funktionen
 Net Search Extender-Suche 267

G

GPP, Dokumentmodell
 Details 187
 Dokumenttypdefinition 199
 Elementparameter 187
 GPP-Dokumente indexieren 188

H

HELP, Befehl
 DB2 Net Search Extender 251
Hilfe
 SQL-Anweisungen 322
Hochverfügbarkeit
 Net Search Extender 41
HTML, Dokumentmodell-DTD 199

I

Indizes

- Kurznamen 11
- Löschen von Ereignissen 119

Informationskataloge

- Sichten
 - Informationen auf Datenbankebene 285
 - Informationen auf Indexebe­ne 288
 - Textindextabellen 292
- Übersicht 285

Installation

- Net Search Extender
 - Befehl db2nse_install 253
 - Befehl db2nsels 254
 - Dateinamen 33
 - Filtersoftware Outside In 35
 - partitionierte DB2-Server 27
 - Systemvoraussetzungen 25
 - Überprüfung (UNIX) 37
 - Überprüfung (Windows) 37
 - UNIX 29
 - Verzeichnisnamen 33
 - Windows 31

Instanzservices

- Aktualisierungsservices 89
- Sperrenservices 85
- starten 83
- stoppen 83
- Übersicht 10

K

Kompilierung

- Thesaurus 163

L

Leistung

- Indizes
 - Empfehlungen 109
 - Ursachen für Beeinträchtigung 109

Lernprogramme

- Fehlerbehebung 327
- Fehlerbestimmung 327
- Liste 327
- pureXML 327

M

Migration

- Net Search Extender
 - 32-Bit auf 64-Bit 49
 - Tool 49

N

Net Search Extender

- Attributwerte 186
- Beschädigung von Indexdateien
 - vorbeugende Maßnahmen 71
- Codepages 69
- Datenbanken
 - Aktivierung 93
 - Inaktivierung 95
- Dienstprogramm zur Indexprüfung 123

Net Search Extender (Forts.)

- Dokumentformate 69
 - erweiterte textverwaltete Zwischenspeicherinfrastruktur 77
 - High Availability Disaster Recovery 41
 - Indexmigration 53
 - Indexnamen 67
 - Indexspeicher 59
 - Installation 23
 - Instanzservices 10, 81
 - Outer Joins 153
 - Planung 57
 - Prüfung eines Textindex 121
 - Ressourcen 61
 - Sonderzeichen als Teil eines Worts 173
 - Spaltennamen 67
 - Stoppwörter 172
 - Tabellenamen 67
 - Teilaktualisierungen 77
 - Textsuchsteuerkomponente 171
 - Tipps 313
 - Tokenanalyse 171
 - Übersicht 3, 17
 - Unterstützung für separate abgeschirmte Benutzer 14
 - Upgrade 45
 - Ursachencodes 297
 - Verwaltungstabellen
 - Sichten 10
 - Übersicht 10
 - Verzeichnisposition 59
 - XML-Unterstützung 12
 - Zeichennormalisierung 173
- NUMBEROFMATCHES, Funktion 268

O

Outer Joins

- Textsuche 153

Outside In-Filter

- Attributwerte 203
- Details 196
- Indexierung 197

P

Partitionierte Datenbanken

- Net Search Extender 10

Partitionierte Tabellen

- Übersicht 12

R

Reorganisation

- Textindizes 113

Replikation

- Erfassungstabellen 227
- Replikationserfassungstabelle 227
- RESET PENDING, Befehl 247

Rückkehrcodes

- Installation 311

S

SCORE, Funktion

- Details 268

- SCORE, Funktion (*Forts.*)
 - Parameter 268
 - Server
 - Tracefunktion zur Fehleranalyse 307
 - Sichten
 - db2ext.textindexes 127
 - Sichten für Net Search Extender
 - Ereignisinformationen 292
 - Informationen auf Datenbankebene
 - db2ext.dbdefaults 285
 - Informationen auf Indexebene
 - db2ext.indexconfiguration 291
 - db2ext.textconfiguration 288
 - db2ext.textindexes 288, 289
 - db2ext.textindexformats 291
 - db2ext.textinformats 288
 - Protokolltabelle 293
 - Skalare SQL-Suchfunktionen
 - CONTAINS 135
 - in mehreren Spalten suchen 151
 - NUMBEROFMATCHES 136
 - SCORE 136
 - SQL-Suchargumente 137
 - Übersicht 6
 - Sonderzeichen als Teil eines Worts verwenden 173
 - Spaltenumsetzungsfunktion 9
 - Speicher
 - gespeicherte Prozeduren 63
 - Voraussetzungen
 - AIX 63
 - Linux 65
 - Solaris 64
 - Windows 64
 - Sperrenservices
 - Anzeige einer Momentaufnahme 87
 - Details 85
 - SQL
 - Tabellenwertfunktion
 - Hervorhebung 147
 - SQL-Anweisungen
 - Hilfe
 - anzeigen 322
 - SQL-Suchargumente
 - Begriffe in Abschnitten 140
 - Begriffe in beliebiger Reihenfolge 137
 - Begriffe in fester Reihenfolge 140
 - Begriffe in gleichem Absatz 140
 - Begriffe in gleichem Satz 140
 - Boolescher Operator AND 137
 - Boolescher Operator NOT 138
 - Boolescher Operator OR 137
 - Escapezeichen in Platzhalterzeichen 139
 - Freitextsuche 142
 - numerische Attribute 141
 - Platzhalterzeichen 139
 - Suche nach groben Übereinstimmungen 138
 - Thesaurussuche 141
 - START (Net Search Extender-Befehl) 209
 - STOP, Befehl 211
 - Stoppwörter 172
 - Strukturierte Dokumente
 - Details 183
 - Dokumentmodelle 183
 - Net Search Extender 175
 - Suchargument, Schlüsselwörter
 - THESAURUS 260
 - Suchargumente
 - Syntax 257
 - Übersicht 257
 - Suche mit gespeicherten Prozeduren
 - DB2EXT.TEXTSEARCH, Befehl 277
 - Funktion 277
 - Übersicht 7
 - Suchen
 - XML-Dokumentverarbeitung 177
 - Suchfunktionen
 - Beispiele 143
 - CONTAINS 267
 - DB2EXT.HIGHLIGHT 272
 - DB2EXT.TEXTSEARCH 269
 - Details 133
 - gespeicherte Prozeduren 145, 277
 - Leistungsaspekte 155
 - NUMBEROFMATCHES 268
 - SCORE 268
 - skalare SQL-Suchfunktion 135
 - SQL-Tabellenwertfunktion 147
 - Übersicht 267
 - Suchparameter 260
 - Szenarios
 - Net Search Extender
 - Skalare SQL-Suchfunktion 157
 - SQL-Tabellenwertfunktion 160
 - Suche mit gespeicherten Prozeduren 159
 - Übersicht 157
- ## T
- Tabellenwertfunktion
 - Übersicht 8
 - Teilaktualisierungen auf Integritätsverarbeitungsbasis 13
 - Textindizes
 - Aktualisierung 113
 - ändern 117
 - Anzeige des Status 127
 - Backup durchführen 129
 - Datenbank löschen 309
 - Details 111
 - Entfernen von Dateien aus dem Verzeichnis /tmp 131
 - Erstellung
 - Aktivieren einer Datenbank 93
 - bereichspartitionierte Tabellen 107
 - Binärdatentypen 99
 - Inaktivierung einer Datenbank 95
 - Kurznamen 101
 - nicht unterstützte Datentypen 100
 - Suche mit gespeicherten Prozeduren 102
 - Übersicht 91, 97
 - Indexereignisse löschen 119
 - Löschen 125
 - Reorganisation 113
 - Restore durchführen 129
 - Sichten 105
 - Tabelle löschen 309
 - Verwaltung 91
 - Textsuchsteuerkomponente
 - Sonderzeichen als Teil eines Worts verwenden 173
 - Zeichennormalisierung 173
 - Thesauren
 - Definieren von Relationen 163
 - Definitionsdatei erstellen 164
 - Erstellung 163
 - Fehlernachrichten 168

Thesauren (*Forts.*)
 Kompilierung 163
 Kompilierung einer Definitionsdatei 165
 Struktur 161
 Suchsyntax 260
 THESAURUS, Schlüsselwort 260
 Übersicht 161
 Unterstützung 165
 vordefinierte Relationen 162
Thesaurus
 Unterstützte CCSIDs 167
Thesaurussuche
 Syntax 260
 THESAURUS, Schlüsselwort 260
Tokenanalyse 171
Tracefunktion zur Fehleranalyse 307

XML-Dokumentverarbeitung (*Forts.*)
 XQuery-Unterstützung 179

Z

Zeichennormalisierung 173

U

UPDATE INDEX (Net Search Extender-Befehl) 248

V

Verwaltungsbefehle für Net Search Extender

Datenbankadministratoren
 DB2EXTHL 217
 DISABLE DATABASE 215
 ENABLE DATABASE 213
 Zusammenfassung 213

Instanzeigner
 CONTROL 207
 START 209
 STOP 211
 Zusammenfassung 207

Texttabelleneigner
 ACTIVATE CACHE 220
 ALTER INDEX 221
 CLEAR EVENTS 225
 COPYRIGHT 252
 CREATE INDEX 227
 DB2EXTTH 246
 DEACTIVATE CACHE 242
 DROP INDEX 244
 HELP 251
 RESET PENDING 247
 UPDATE INDEX 248
 Zusammenfassung 219

W

Windows
 Systemfehler 281

X

XML-Dekomposition
 Net Search Extender 186
XML-Dokumentmodell
 Details 191
 Dokumenttypdefinition 199
 XPath-Ausdrücke, Semantik 200
XML-Dokumentverarbeitung
 Abschnittssuche 177
 angepasstes XML-Dokumentmodell 178
 XML-Standarddokumentmodell 177



SC12-4689-01



Spine information:

IBM DB2 10.1 for Linux, UNIX and Windows

Net Search Extender - Verwaltung und Benutzerhandbuch

