

IBM DB2 Universal Database



Net Search Extender
管理およびユーザース・ガイド

バージョン 8.2

IBM DB2 Universal Database



Net Search Extender
管理およびユーザース・ガイド

バージョン 8.2

ご注意!

本書および本書で紹介する製品をご使用になる前に、『特記事項』に記載されている情報をお読みください。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： SH12-6740-04
IBM DB2 Universal Database
Net Search Extender
Administration and User's Guide
Version 8.2

発 行： 日本アイ・ビー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

第1刷 2004.8

この文書では、平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、
平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 1995, 2004. All rights reserved.

© Copyright IBM Japan 2004

目次

本書について	vii
本書の対象読者	vii
本書の使用方法	vii
構文図の読み方	viii
関連情報	viii
IBM と連絡をとる	ix
製品情報	ix

第 1 部 ユーザーズ・ガイド 1

第 1 章 概要と概念 3

主要な概念	3
SQL スカラー検索関数の使用	5
ストアド・プロシージャ検索の使用	6
検索での SQL 表値関数の使用	7
追加概念	8
列変換関数	8
インスタンス・サービス	9
外部保管データ	9
管理ビュー	9
パーティション・データベース・サポート	9
フェデレーテッド・データベースのニックネームに 対する索引	9
主要な機能	10
DB2 クライアント/サーバー環境内の DB2 Net Search Extender	11

第 2 章 インストール 13

システム要件	13
パーティション化された DB2 サーバーのインス トールの概要 (UNIX)	14
UNIX でのインストール	14
UNIX 用のステップ 1: 製品コンポーネントのイ ンストール	15
UNIX 用のステップ 2: DB2 インスタンスの更新	15
Windows でのインストール	15
ディレクトリー名とファイル名	16
Outside-In ライブラリーのインストール	16
インストール検査	16
Windows でのインストール検査	16
UNIX でのインストール検査	17

第 3 章 ユーザー・シナリオ 19

SQL スカラー検索関数の場合の簡単な例	19
キャッシュを使用したストアド・プロシージャ 検索の場合の簡単な例	21
SQL 表値関数の場合の簡単な例	22

第 4 章 計画 23

ディレクトリーのロケーションおよび索引ストレ ージ	23
文書フォーマットおよびサポートされるコード・ベ ージ	23
Outside-In フィルター・ソフトウェア	24
ユーザーの役割	25
索引作成でのコマンド行または DB2 コントロー ル・センター・インターフェースの使用	26

第 5 章 Net Search Extender インスタ ンス・サービス 27

DB2 Net Search Extender の開始および停止	27
ロッキング・サービス	27
ロッキング・サービスの使用	27
ロック・スナップショットの表示	29
更新サービス	30

第 6 章 テキスト索引の作成と保守 31

db2text コマンドの紹介	31
データベースを使用可能にする	32
データベースを使用不可にする	33
テキスト索引の作成	35
バイナリー・データ・タイプのテキスト索引の作 成	36
サポートされないデータ・タイプのテキスト索引 の作成	37
DATALINK データ・タイプのテキスト索引の作 成	38
データ・リンク jar ファイルのインストール	38
DB2 レプリケーションで、インクリメンタル索引 更新による、ニックネームのテキスト索引を作成 する	38
ストアド・プロシージャ検索で使用できるテ キスト索引の作成	40
ビューに基づくテキスト索引	43
パフォーマンスに関する考慮事項	45
テキスト索引の保守	46
テキスト索引の更新と再編成	46
テキスト索引の変更	48
索引イベントのクリア	48
テキスト索引のドロップ	49
テキスト索引状況の表示	49
索引のバックアップとリストア	50
/tmp ディレクトリーからのファイルの除去	50

第 7 章 DB2 コントロール・センターの 使用 51

DB2 Net Search Extender インスタンス・サービスの 開始および停止	52
データベースを使用可能にする	53

テキスト索引の管理	54
テキスト索引の作成	55
テキスト索引の保守	66

第 8 章 検索 73

SQL スカラー検索関数を使用するテキストの検索	74
照会を行う	74
検索を実行し、検出された一致の数を戻す	74
検索を実行し、検出されたテキスト文書のスコアを戻す	75
SQL 検索引き数の指定	75
任意の順序での用語検索	75
ブール演算子 AND と OR を使用する検索	76
ファジー検索	76
用語の一部の検索 (文字のマスキング)	77
マスク文字をすでに含んでいる用語の検索	77
固定順序での用語検索	77
同じ文または同じ段落内での用語検索	78
構造化文書のセクション内での用語検索	78
ブール演算子 NOT を使用する検索	78
シソーラス検索	78
数値属性検索	79
フリー・テキスト検索	79
その他の検索構文の例	80
ストアード・プロシージャ検索を使用するテキスト検索	80
SQL 表値関数を使用する検索	81
HIGHLIGHT 関数の使用	82
複数の列の検索	84
パフォーマンスに関する考慮事項	84

第 9 章 構造化文書を使用する作業 87

第 10 章 検索項目を拡張するためのシソーラスの使用 89

シソーラスの構造	89
定義済みシソーラス関係	90
ユーザー独自の関係の定義	91
シソーラスの作成およびコンパイル	91
シソーラス定義ファイルの作成	91
定義ファイルのシソーラス辞書へのコンパイル	93

第 2 部 参照情報 95

第 11 章 インスタンス所有者用の管理コマンド 97

CONTROL	98
START	100
STOP	101

第 12 章 データベース管理者用の管理コマンド 103

ENABLE DATABASE	104
DISABLE DATABASE	106
DB2EXTDL (ユーティリティ)	108

DB2EXTHL (ユーティリティ)	109
--------------------	-----

第 13 章 テキスト表所有者用の管理コマンド 111

ACTIVATE CACHE	112
ALTER INDEX	114
CLEAR EVENTS	118
CREATE INDEX	120
DEACTIVATE CACHE	135
DROP INDEX	137
DB2EXTTH (ユーティリティ)	139
UPDATE INDEX	141
HELP	145
COPYRIGHT	146

第 14 章 検索引き数の構文 147

検索引き数	148
-------	-----

第 15 章 SQL スカラー検索関数および SQL 表値関数 155

検索関数のサマリー	155
CONTAINS	156
NUMBEROFMATCHES	157
SCORE	158
DB2EXT.TEXTSEARCH	159
DB2EXT.HIGHLIGHT	163

第 16 章 ストアード・プロシージャ検索関数 167

DB2EXT.TEXTSEARCH (ストアード・プロシージャ検索用)	168
-------------------------------------	-----

第 17 章 構造化文書のサポート 171

文書モデルが構造化文書を記述する方法	171
文書モデルの例	171
文書モデル	172
テキスト・フィールドおよび文書属性	173
デフォルトの文書モデル	174
構造化プレーン・テキストの場合の文書モデルの定義	175
エレメント・パラメーター	176
GPP 文書を索引付けするとどうなるか	177
HTML 文書の場合の文書モデルの定義	177
エレメント・パラメーター	178
XML 文書の場合の文書モデルの定義	179
エレメント・パラメーター	180
XML 文書を索引付けするとどうなるか	183
Outside-In フィルター文書の場合の文書モデルの定義	184
エレメント・パラメーター	184
Outside-In 文書を索引付けするとどうなるか	185

第 18 章 シソーラス・サポート 187

第 3 部 付録 189

付録 A. 移行 191

Net Search Extender バージョン 8.1 から Net Search Extender バージョン 8.1.x および 8.2 への移動 192

Net Search Extender バージョン 7.2 から Net Search Extender バージョン 8.1.x への移動 192

テキスト情報エクステンダー バージョン 7.2 から Net Search Extender バージョン 8.1.x への移動 193

付録 B. 大量のメモリーの使用 195

AIX (32 ビットおよび 64 ビット) 195

Windows (32 ビット) 196

Solaris オペレーティング環境 (32 ビットおよび 64 ビット) 196

Linux 197

HP-UX (32 ビットおよび 64 ビット) 197

付録 C. Net Search Extender インフォメーション・カタログ 199

データベース・レベル情報のビュー 199

 db2ext.proxyinformation 表 201

索引レベル情報のビュー 201

 db2ext.textindexes ビュー 202

 db2ext.indexconfiguration ビュー 203

 db2ext.textindexformats ビュー 204

テキスト索引の表ビュー 205

 イベント・ビュー 205

 ログ表、ビュー、およびニックネーム 206

付録 D. サポートされている CCSID 209

CCSID 209

付録 E. サポートされている言語 215

付録 F. Net Search Extender メッセージ 217

通知および警告メッセージ 217

エラー・メッセージ 218

付録 G. 文書モデルの参照情報 239

文書モデル用の DTD 239

ロケーター (XPath) 式のセマンティクス 240

テキスト・フィールドおよび文書属性の制限 242

Outside-In タグ属性値 243

付録 H. テキスト検索エンジン 245

トークン化 245

 ワード 245

 文 245

 段落 245

ストップワード 245

 ストップワードをサポートする言語 246

構成 246

 文字正規化 246

 ワードの一部としての特殊文字の使用 247

付録 I. テキスト検索エンジン理由コード 249

付録 J. トラブルシューティング 257

障害のトレース 257

正しい Net Search Extender コマンドを使用しない DB2 オブジェクトのドロップ 257

 表のドロップ 257

 データベースのドロップ 258

付録 K. データ・リンク・メッセージ 259

付録 L. シソーラスでサポートされている CCSID 261

CCSID 261

付録 M. シソーラス・ツールが戻すメッセージ 263

付録 N. Windows システム・エラー 265

システム・エラー 265

付録 O. 特記事項 269

商標 271

用語集 273

索引 275

本書について

本書は、IBM DB2 Universal Database™ Net Search Extender を使用して、テキスト・データの検索用に DB2® データベースを準備し、保守する方法について説明します。また、提供されている SQL 関数を使用して、このようなタイプのデータにアクセスし、操作する方法についても説明します。DB2 Net Search Extender の関数をユーザー・プログラムの SQL ステートメントに組み込むことによって、強力で多様なテキスト検索プログラムを作成できます。

本書での「DB2」は DB2 UDB を指します。

本書の対象読者

本書の対象読者は、DB2 管理の概念、ツール、および手法についての知識がある DB2 データベース管理者です。

本書は、SQL および DB2 アプリケーション・プログラムに使用できる 1 つまたは複数のプログラム言語の知識がある、DB2 アプリケーション・プログラマーも対象にしています。

本書の使用方法

本書の構成は次のとおりです。

『第 1 部 ユーザーズ・ガイド』

第 1 部では、DB2 Net Search Extender について概説し、インストールおよびセットアップの方法について説明し、プランに関する考慮事項を示します。さらに、テキストを検索できるように、DB2 データベースを準備し、保守する方法についても説明します。

DB2 Net Search Extender についての知識がなく、テキストを検索するために、DB2 Net Search Extender の関数を使用する方法について学習したい場合には、第 1 部をお読みください。

『第 2 部 参照情報』

第 2 部では、DB2 Net Search Extender 関数およびコマンドに関する参照情報を示します。

DB2 Net Search Extender の概念およびタスクについての知識はあるものの、特定の DB2 Net Search Extender の関数およびコマンドに関する情報が必要な場合に、第 2 部をお読みください。

『第 3 部 付録』

第 3 部では、DB2 Net Search Extender に関する追加の参照情報を示します。これには、移行、メモリーの使用、ビュー、文書モデル、メッセージ、およびコードに関する情報が含まれています。

DB2 Net Search Extender に関する特定の参照情報が必要な場合に、第 3 部をお読みください。

構文図の読み方

本書では、以下の定義に従って構文図が記載されています。

- 構文図は、左から右、上から下に、線に沿って読みます。

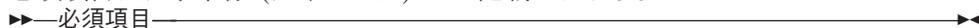
記号  は、ステートメントの始まりを示します。

記号  は、ステートメントの構文が次の線に続くことを示します。

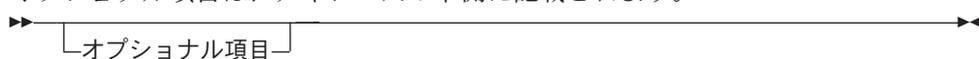
記号  は、ステートメントが前の線からの続きであることを示します。

記号  は、ステートメントの終わりを示します。

- 必須項目は、水平線 (メインパス) 上に記載されます。

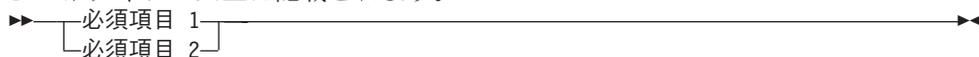


- オプション項目は、メインパスの下側に記載されます。



- 2 つ以上の項目からの選択が可能な場合は、複数の項目が縦に並んで記載されません。

項目から 1 つを選択しなければならない場合は、併記されている項目のうちの 1 つがメインパス上に記載されます。



どの項目も選択しないことが可能な場合は、縦に並べられた項目全体がメインパスより下側に表示されます。



縦に並んだ選択項目の上側に反復矢印記号がある場合は、その中から複数の項目を繰り返し選択できるということです。



- キーワードは英大文字で示します。表示されているとおりに入力する必要があります。変数は英小文字で示しています (例: srcpath)。それらの変数は、構文中でユーザーが指定する名前または値です。
- 句読点、括弧、算術演算子、その他の記号が示されている場合は、それをそのとおりに構文の一部として入力する必要があります。

関連情報

IBM DB2 Universal Database バージョン 8

- 「*IBM DB2 Universal Database 概説およびインストール バージョン 8 DB2 サーバー版 (GC88-9148)、DB2 クライアント版 (GC88-9144)、DB2 Connect Personal Edition 版 (GC88-9146)、DB2 Personal Edition 版 (GC88-9150)、および IBM Data Links Manager 版 (GC88-9141)*」。これらの資料は、該当するプラットフォームでの DB2 Universal Database のインストール、構成、移行の計画について説明しています。
- 「*IBM DB2 Universal Database 管理ガイド バージョン 8 プランニング (SC88-9135)、パフォーマンス (SC88-9134)、およびインプリメンテーション (SC88-9133)*」。これらの資料は、DB2 データベースを設計し、インプリメントする方法について説明しています。
- 「*IBM DB2 Universal Database コール・レベル・インターフェース ガイドおよびリファレンス、第 1 巻 バージョン 8 (SC88-9159)*」。この資料は、Microsoft ODBC の仕様との互換性を持つ呼び出し可能 SQL インターフェースの DB2 コール・レベル・インターフェースを使用して、DB2 データベースにアクセスするアプリケーションを開発する方法について説明しています。
- 「*IBM DB2 Universal Database コール・レベル・インターフェース ガイドおよびリファレンス、第 2 巻 バージョン 8 (SC88-9160)*」。この資料は、Microsoft ODBC の仕様との互換性を持つ呼び出し可能 SQL インターフェースの DB2 コール・レベル・インターフェースを使用して、DB2 データベースにアクセスするアプリケーションを開発する方法について説明しています。
- 「*IBM DB2 Universal Database コマンド・リファレンス バージョン 8 (SC88-9140)*」。この資料は、DB2 コマンド行プロセッサを使用する方法について説明し、DB2 コマンドに関する参照情報を示しています。
- 「*IBM DB2 Universal Database レプリケーションのガイドおよびリファレンス バージョン 8 (SC88-9163)*」。この資料は、DB2 データ・レプリケーション環境を計画、セットアップ、および保守する方法を説明しています。

IBM DB2 Universal Database Enterprise Server Edition バージョン 8

- 「*IBM DB2 Connect Enterprise Edition 概説およびインストール バージョン 8 (GC88-9145)*」。この資料は、該当するプラットフォームでの DB2 Universal Database Enterprise-Extended Edition のインストール、構成の計画について説明しています。

IBM と連絡をとる

技術上の問題がある場合は、お客様サポートにご連絡ください。

製品情報

DB2 Universal Database 製品に関する情報は、www.ibm.com/software/data/db2/udb から入手できます。

このサイトには、技術ライブラリー、資料の注文方法、クライアント・ダウンロード、ニュースグループ、フィックスバック、ニュース、および Web リソースへのリンクに関する最新情報が掲載されています。

米国以外の国で IBM に連絡する方法については、IBM Worldwide ページ (www.ibm.com/planetwide) にアクセスしてください。

第 1 部 ユーザーズ・ガイド

第 1 章 概要と概念

DB2 Net Search Extender バージョン 8.1 は、DB2 エクステンダー (DB2 Extenders™) ファミリーに含まれます。

DB2 Net Search Extender バージョン 8.1 は、DB2 テキスト情報エクステンダー バージョン 7.2、および Net Search Extender バージョン 7.2 に代わり、DB2 やその他のデータベース、およびファイル・システムに保管されているフルテキスト文書を SQL 照会を使用して検索するための、高速、多様でインテリジェントな方法をユーザーとアプリケーション・プログラマーに提供します。

主要な概念

DB2 Net Search Extender の機能を十分に理解するには、このセクションで**太字**で示されている主要な用語、および使用可能なさまざまなオプションを理解する必要があります。DB2 Universal Database の概念および用語の基礎的な理解も必要です。

基本的に、DB2 Net Search Extender は、データベース表の列に含まれている**テキスト文書**の検索を行います。

テキスト文書は、一意に識別可能でなければなりません。この目的のために、Net Search Extender は表の**主キー**を使用します。

文書は、HTML や XML などさまざまな形式をとることができます。

膨大な時間を必要とするテキスト文書の順次検索の代わりに、Net Search Extender は、文書の検索を可能にする**テキスト索引**を作成します。

テキスト索引は、テキスト文書から抽出された重要な**用語**で構成されます。

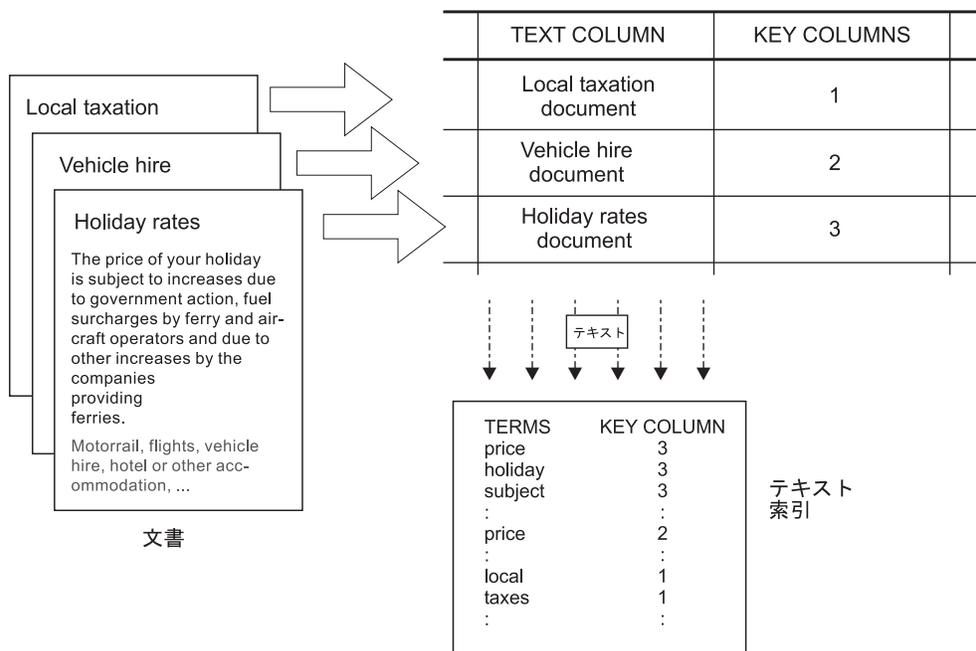


図1. テキスト索引の作成

テキスト索引作成は、索引のロケーションのような、索引のプロパティの定義および宣言を行う処理です。作成後、テキスト索引にはデータは含まれていません。**索引更新**は、テキスト索引にデータを追加する処理です。最初の索引更新で、テキスト列のすべてのテキスト文書が索引に追加されます。これは、**初期更新**と呼ばれます。

検索にテキスト索引を使用することによって、表とテキスト索引との間の同期を考慮する必要がでてきます。理由は、テキスト文書の追加、削除、および更新などの表に対する後からの変更が、テキスト索引に反映される必要があるためです。

Net Search Extender での同期処理は、新規の文書、変更や削除された文書に関する情報を自動的に**ログ表**に保管する**トリガー**に基づいて行われます。それぞれのテキスト索引ごとに1つのログ表が存在します。ログ表の内容を対応するテキスト索引に適用することを**インクリメンタル更新**と呼びます。

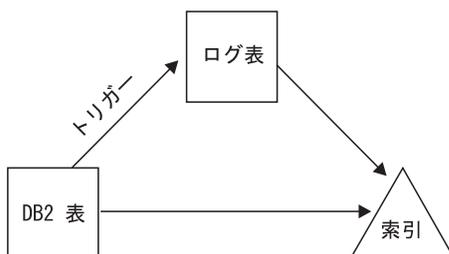


図2. インクリメンタル更新処理

手動または**自動**オプションを使用して、テキスト索引を更新できます。自動オプションは、日付および時刻を設定する更新スケジュールを使用します。

これらのどちらのオプションを使用しても、テキスト文書の更新、削除、および挿入を行うトランザクションの範囲内で、テキスト索引が同期化されることはありません。Net Search Extender の非同期テキスト索引作成を使用すると、パフォーマンスおよび並行性が向上します。別々のトランザクション内で、索引の非常に小さな部分のコピーに対して更新が行われます。オリジナルに代わってコピーが置かれる非常に短い間だけ、索引に読み取りアクセスのロックがかけられるに過ぎません。検索操作ではこのことは意識されません。詳しくは、27 ページの『第 5 章 Net Search Extender インスタンス・サービス』を参照してください。

テキスト索引は、索引ファイルのロケーションおよび自動更新プロパティーなどの、特定のプロパティーをもっています。必要なら、プロパティーの一部を変更できます。これは索引の**変更**と呼ばれます。

そのようなプロパティーの 1 つに、ORDER BY 句で表の列のテキスト索引を事前ソートするかどうかというものがあります。このような場合、初期更新を行うと、指定した順序でテキスト文書に索引が付けられて、検索結果がこの順序で戻されます。

たとえば、事前ソートされた本を指定すると、本の価格に応じて抽出が行われます。リレーショナル・データベース・システムに関する本で一番安いものを探す場合は、テキスト検索を制限して一番安い最初の数冊の本だけを戻すようにすることができます。ただし、事前ソートされた索引がないと、すべての本を検索し、これらの本と一番安い本とを結合する必要があります。この場合は、操作に多くの時間がかかります。

Net Search Extender では、テキスト列ごとに事前ソート済み索引を複数もつことができます。たとえば、1 つ目の索引で、出版の日付に従って本の事前ソートを行い、2 つ目の索引で、価格に従って本の事前ソートを行うことができます。

通常、テキスト索引を作成した後の最初の更新が初期更新になり、それ以降はインクリメンタル更新になります。ただし、事前ソート済み索引を使用して作業を行う場合は、更新の際に順序を維持する必要があります。**Recreate Index on Update** オプションを使用すると、この処理を行うことができます。このオプションでは、更新が行われるたびに索引がすべて再作成されます。

テキスト索引の更新が終わると、以下のいずれかのオプションを使用して、検索を行うことができます。

- SQL スカラー検索関数
- ストアード・プロシージャ検索
- SQL 表値関数

検索オプションにはいろいろな操作特性があります。以下のセクションでこれらのオプションを説明します。

SQL スカラー検索関数の使用

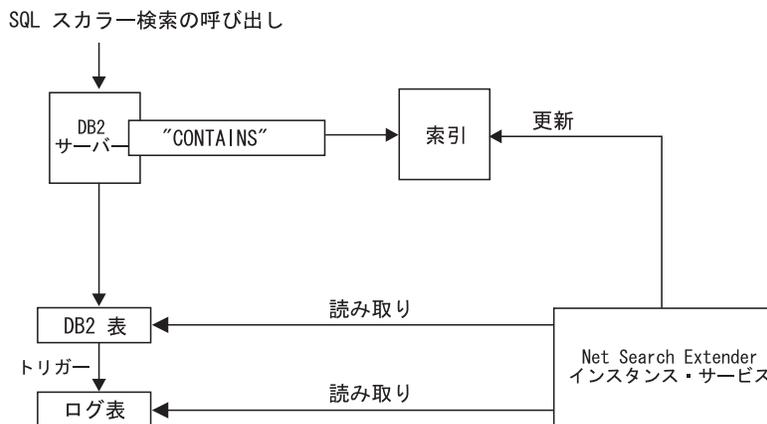


図3. 検索での SQL スカラー検索関数の使用

Net Search Extender には、SQL 内でシームレスに統合される 3 つのスカラー・テキスト検索関数 (CONTAINS、NUMBEROFMATCHES、および SCORE) があります。検索関数は、SQL 照会内で標準の SQL 式を使用する場合と同じ場所で使用できます。典型的な照会は以下のとおりです。

```
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
```

SQL スカラー関数は、テキスト文書が、指定されたテキスト検索条件にどの程度一致したかを示す指標のみを戻します。SQL 照会の SELECT 段階で、エンド・ユーザーに戻す情報が決定されます。

SQL スカラー検索関数を、デフォルトの検索方式として使用してください。これらの検索関数は、ほとんどの状況に適合するはずですが、テキストの検索式が他の異なる条件と結合している場合には、特にそう言えます。

DB2 のオプティマイザーは、CONTAINS 述部に一致するテキスト文書の個数を予測し、代わりに別のアクセス・プランで、どの程度の時間が必要になるかを認識できます。オプティマイザーは、最短の時間のアクセス・プランを選択します。

ストアド・プロシージャ検索の使用

ストアド・プロシージャ検索は、SQL スカラー検索関数とは異なる動作をします。テキスト索引の作成時に、表またはビューのどの列をエンド・ユーザーに戻すかを指定する必要があります。このデータはメイン・メモリーのキャッシュに保管されます。このようにすると、ストアド・プロシージャ検索で、非常にすばやく検索結果を戻すことができます。キャッシュは、使用する前にアクティブ化する必要があります。また、対応する非アクティブ化コマンドがあります。

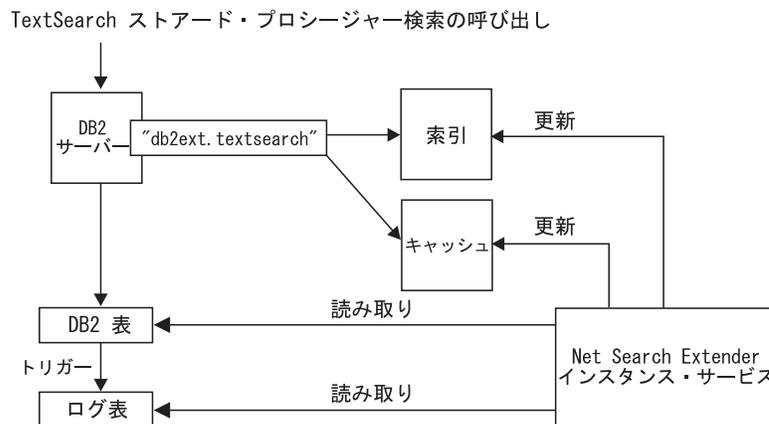


図 4. ストアド・プロシージャ検索の使用

ACTIVATE コマンドを使用すると、データは一時キャッシュ (アクティブ化の際に最初から作成される) か、ディスク上に保持される永続キャッシュにロードされます。

検索にストアド・プロシージャを使用すると、索引の更新にどれだけのメモリーが必要か、およびどれだけの空きメモリーが残っていないかなどのメモリー計算が必要になります。

ストアド・プロシージャは、ビュー上に作成されたテキスト索引を処理できます。ただし、ビュー上にトリガーを作成できないため、すべての変更が自動的に認識されるわけではありません。変更されたこの情報は、手動でログ表に追加するか、RECREATE オプションを使用して作業することができます。

ストアド・プロシージャ検索は、テキスト検索のみの照会を行うハイパフォーマンスでスケーラビリティの高いアプリケーションに使用してください。たとえば、テキスト検索結果とその他の複雑な SQL 条件の結果を結合する必要のない検索です。

SQL スカラー検索関数との主な機能上の相違点は以下のとおりです。

- ストアド・プロシージャ検索は、任意の SQL 照会には使用できず、定義済みのキャッシュ表に対する照会である。
- ストアド・プロシージャ検索は、ビューの索引を活用することができる。
- ストアド・プロシージャ検索は、列の事前ソートされた複数のテキスト索引を利用できる。

このオプションには大量のメイン・メモリーが必要です。詳細に関しては、195 ページの『付録 B. 大量のメモリーの使用』を参照してください。

検索での SQL 表値関数の使用

主要な概念

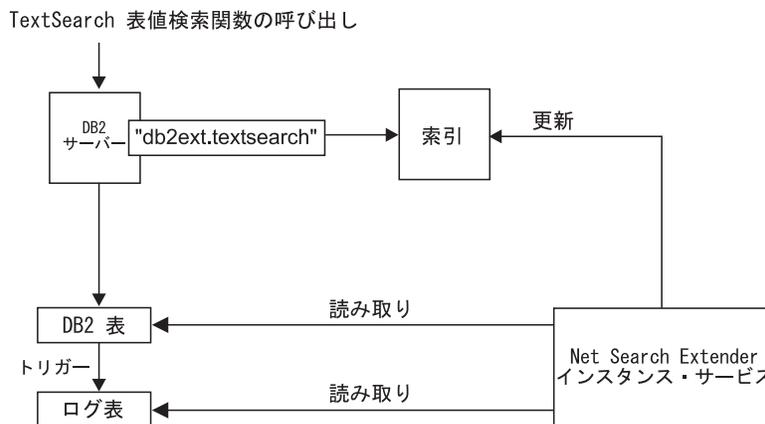


図5. 検索での SQL 表値関数の使用

SQL 表値関数は、SQL スカラー検索関数とストアード・プロシージャ検索の中間的な方式です。SQL 表値関数とともに `db2ext.highlight` 関数を使用することにより、検索結果としてその文書が選ばれた理由に関する情報を入手することができます。

ストアード・プロシージャ検索との主な機能上の相違点は以下のとおりです。

- キャッシュが必要ない (キャッシュが利用されない)。
- 任意の SQL ステートメントに表値関数を使用できる。
- 大量のメイン・メモリーは必要ない。

SQL スカラー検索関数との主な機能上の相違点は以下のとおりです。

- SQL 表値関数はビューの索引を利用できる。

SQL 表値関数は、通常 SQL スカラー関数を使用する場合で、ビューのテキスト索引も使用したい場合に使用してください。

追加概念

DB2 Net Search Extender の主要な概念を理解するだけでなく、その他のいくつかの概念も理解する必要があります。

Net Search Extender に基づくアプリケーション開発の詳細については、23 ページの『第 4 章 計画』を参照してください。

列変換関数

ユーザー独自の関数を使用して、サポートされていないフォーマットまたはデータ・タイプを、サポートされている形式またはデータ・タイプに変換することができます。ユーザー定義関数 (UDF) を指定することによって、オリジナルのテキスト文書を入力として入手できます。UDF からの出力は、索引作成の際に処理できるサポートされたフォーマットになります。

サポートされていない外部データ・ストアに保管されている文書の索引作成にも、この機能を使用できます。この場合、DB2 列には文書参照が含まれているため、関数は、適切な文書参照をもった文書の内容を戻します。

37 ページの『サポートされないデータ・タイプのテキスト索引の作成』を参照してください。

インスタンス・サービス

Net Search Extender インスタンス・サービスは、索引の特定のロッキング・サービスおよびテキスト索引の更新サービス (自動および手動の両方) を行います。

詳細については、27 ページの『第 5 章 Net Search Extender インスタンス・サービス』を参照してください。

外部保管データ

ほとんどの場合、テキスト索引が作成されるデータは、CLOB または VARCHAR 型などの、ネイティブ DB2 表の列内に保管されています。

ただし、ファイルまたはその他のデータベースなどの外部に保管されるテキスト文書もサポートされています。ファイルに保管される文書の場合、DB2 データ・リンク機能を使用できます。その他のデータベースに保管される文書の場合、テキスト索引を作成する DB2 ニックネーム表を使用してください。

viii ページの『関連情報』を参照してください。

サポートされていない外部データ・ストアに保管されているデータに対して、列変換関数を使用することもできます。8 ページの『列変換関数』を参照してください。

管理ビュー

DB2 Net Search Extender では、いくつかのビューを使用できます。それらのビューは、テキスト索引およびそれらのプロパティに関する情報を提供します。

詳細については、199 ページの『付録 C. Net Search Extender インフォメーション・カタログ』を参照してください。

パーティション・データベース・サポート

DB2 Net Search Extender の検索機能は、パーティション・データベース・サポートを次の方法で使用します。

- SQL スカラー関数は、パーティション表に作成された索引を使用する。
- ストアド・プロシージャ検索および SQL 表値関数は、パーティション化された環境の 1 つのノード上の表のみを使用する。

フェデレーテッド・データベースのニックネームに対する索引

フェデレーテッド・データベース内のニックネームにもテキスト索引を作成することができます。これは、リモート・データベース内の表を指し示すものとなります。この場合、ログ表が担う役割 (インクリメンタル索引更新の場合) は、通常の表の索引の場合と異なります。通常の表とは異なり、DB2 トリガーをニックネームに対して作成することはできません。したがって、トリガーを使用して文書の変更情報をログ表に挿入することはできません。そのため、インクリメンタル更新でニックネームの索引を作成するには、以下に挙げる 2 つの異なる方法があります。

- ログ表をフェデレーテッド・データベース内でローカルに作成し、ログ表にニックネームに関する正しい変更情報を記録する責任をアプリケーションに負わせる方法。DB2 ビューの場合、これはインクリメンタル索引更新と似ています。このオプションはデフォルト・オプションです。
- ニックネームによって参照される表に加えられた変更を、いわゆる「変更データ表」(CD 表: DB2 リモート・データベースの場合)、または「整合変更データ表」(CCD 表: DB2 以外のリレーショナル・データベースの場合) にキャプチャーするよう、DB2 レプリケーションをセットアップしておく方法。そうすれば、DB2 Net Search Extender は、ニックネームの索引用にログ表を作成する代わりに、CD または CCD 表を使用できます。この場合、DB2TEXT CREATE INDEX コマンドにキャプチャー表の特性を指定する必要があります。

主要な機能

DB2 Net Search Extender バージョン 8.1 には、次の主要な機能が備わっています。

- 索引付け
 - 大規模データ・ボリュームの高速索引付け
 - 索引の動的更新
 - 索引作成時にメイン・メモリーに表の列を保管することによる、検索時の過度の物理読み取り操作の回避
 - コマンド行、あるいは DB2 コントロール・センターから索引の作成が可能
 - HTML、XML などのいろいろなテキスト・フォーマットのサポート
 - サード・パーティーのフィルター・ソフトウェア「Outside-In」のサポート
 - ニックネーム表サポート
 - DB2 Data Links Manager サポート
 - 事前ソートされたテキスト索引のサポート
 - パーティション・データベース・サポート
- 検索
 - ブール演算
 - 同じ文または同じ段落内でのワードに対する近接検索
 - 検索語と似たスペルを持つワードを検索する「ファジー」検索
 - ワード全体および単一文字に対する、前部、中間、および後部のマスキングを使用するワイルドカード検索
 - フリー・テキスト検索。特定のテキストを含んでいる文書用に、検索引き数を自然言語で表現できる。
 - 特定の文書が検索結果として選ばれた理由を示すための強調表示関数
 - シソーラス・サポート
 - 文書内のセクションへの検索の限定
 - 数値属性のサポート
 - 多数の並行ユーザーによる、大量のテキスト文書に対する高速検索
- 検索結果
 - 索引作成時に、検索結果のソート方法を指定できる

- 大量のデータの中から検索して、結果リストが大量に及ぶものと予想される場合に、検索結果のサブセットを指定できる
- 検索項目に対して、ヒット・カウントによる制限を設定できる
- DB2 オプティマイザーと結合した組み込み SQL 関数が、予想される検索結果に従って最適なプランを自動的に選択する機能

DB2 クライアント/サーバー環境内の DB2 Net Search Extender

DB2 Net Search Extender 検索機能は SQL に統合化されており、サーバーで実行されます。したがって、テキスト検索照会を出すために、クライアント上に Net Search Extender をインストールする必要はありません。

DB2 Net Search Extender は、クライアント・サイドからサーバーへの管理呼び出しをサポートします。クライアント・サイドおよびサーバー・サイドに DB2 Net Search Extender をインストールするか、あるいは、DB2 コントロール・センターを使用して、クライアント・サイドから DB2 Net Search Extender を管理してください。

第 2 章 インストール

この章では、UNIX[®] および Windows[®] システムに DB2 Net Search Extender をインストールする方法について説明します。

Net Search Extender バージョン 8.1 から 8.2 に移行する場合は、191 ページの『付録 A. 移行』を参照してください。

インストール後、DB2[®] Net Search Extender インストール検査スクリプトを実行してください。

システム要件

DB2 Net Search Extender の実行には、次のバージョンのソフトウェアが必要になります。

- DB2 バージョン 8.1 Fix Pack 4。
- Java[™] ランタイム環境 (JRE)。JRE バージョンは、DB2 のバージョンによって異なります。

DB2 Net Search Extender は、次のオペレーティング・システム上で、使用できます。

Solaris オペレーティング環境:

- Solaris 7
- Solaris 8
- Solaris 9

AIX[®]:

- AIX バージョン 4.3.3。ファイルセット x1C.aix43.rte 6.0.0.5 も必要です。
- AIX バージョン 5.1.0/5.2.0。64 ビット・コード・サポート用に、ファイルセット x1C.aix50.rte 6.0.0.5 以上も必要です。

AIX ファイルセットは <http://techsupport.services.ibm.com/server/fixes> からダウンロードできます。

Windows:

- Windows NT[®] バージョン 4
- Windows 2000
- Windows XP

HP:

- HP-UX 11i

Linux:

- Intel[™] x86 (32 ビット) マシン上の Linux は DB2 Net Search Extender をサポートします。

インストール

- zSeries® マシン上の Linux は DB2 Net Search Extender をサポートします。

注

新規 Linux カーネルおよびディストリビューションの妥当性検査状況は頻繁に更新されます。サポートされる Linux ソフトウェア・レベルの最新情報を入手するには、<http://www.ibm.com/software/data/db2/linux/validate> を参照してください。

AIX、Solaris、および HP-UX の場合、DB2 Net Search Extender は 32 ビット・アプリケーションとしても、64 ビット・アプリケーションとしても使用可能です。Linux の場合、Net Search Extender は Intel x86 上では 32 ビット・アプリケーションとして、zSeries 上では 31 ビット・アプリケーションとして使用可能です。

どの DB2 Net Search Extender オペレーティング・システムでも、最小メモリ要件は 30MB です。DB2 Net Search Extender の一般インストールの最小ディスク・スペースは 50MB です。

DB2 Net Search Extender の最小ソフトウェア要件およびハードウェア要件は、DB2 Universal Database™ バージョン 8.1 の要件と同じです。これらの要件および固有のオペレーティング・システムのパッチについては、「*IBM Universal Database 概説* およびインストール バージョン 8」の対応するセクションを参照してください。

追加のハードウェア要件は、選択するテキスト索引のサイズおよびタイプによって異なります。IBM の資料については、viii ページの『関連情報』を参照してください。

パーティション化された DB2 サーバーのインストールの概要 (UNIX)

すべてのノードで、DB2 が正しくインストールおよび構成されていることを確認してください。DB2 をインストールした後、各コンピューターに DB2 Net Search Extender をインストールする必要があります。

注

インスタンス所有者 ID と異なる fenced ユーザー ID では、パーティション・データベースを扱う作業ができません。

UNIX でのインストール

UNIX にインストールするには、次のステップに従ってください。

1. 製品をインストールします。
2. DB2 インスタンスを更新します。

UNIX 用のステップ 1: 製品コンポーネントのインストール

UNIX にインストールするには、次のステップに従ってください。

1. ターゲット・マシンに root ユーザーとしてログオンします。
2. ご使用のプラットフォームの適切なディレクトリーに移動します。
 - `cd /<cdrom>` ここで、`<cdrom>` は CD-ROM ドライブのパスです。
 - `cd /<platform>`
3. `./nsetup.sh` を呼び出して、画面に表示される指示に従います。注: 「Exited with」 メッセージは無視してください。

UNIX 用のステップ 2: DB2 インスタンスの更新

DB2 インスタンスを更新するには、次のステップに従ってください。

1. root ユーザーとしてアクティブであることを確認します。
2. プラットフォームに応じて、以下のいずれかのコマンドを使用します。
 - AIX の場合: `cd /usr/opt/db2_08_01/instance`
 - Solaris、Linux、HP-UX の場合: `cd /opt/IBM/db2/V8.1/instance`
3. `./db2iupdt <db2instance>` を使用して `db2iupdt` を実行します。ここで、`<db2instance>` は、Net Search Extender で使用する既存の DB2 インスタンスのユーザー ID です。
4. ログアウトします。

注

Net Search Extender は、`db2icrt` の実行中に DB2 インスタンスを自動的に新規作成します。

Windows でのインストール

Windows でインストールするには、管理権限を持つユーザー ID でログオンした後、次のステップに従う必要があります。

1. `<cdrom>:\windows\install\setup.exe` を使用して、パッケージからターゲット・マシンにファイルを転送します。すべての DB2 サービスで、ユーザー ID およびパスワードを入力して適切な DB2 Net Search Extender サービスを作成する必要があります。
2. データ転送後にシステムをリブートします。
3. `db2text start` を呼び出して、DB2 Net Search Extender のインスタンス・サービスを開始します。

注

各 DB2 インスタンスが、それぞれ Windows サービスを作成します。DB2 インスタンス・サービスの実行は、システム・アカウントではなく、必ずユーザー・アカウントのもとで行ってください。

DB2 Net Search Extender は Microsoft® Cluster Server ではサポートされていません。

ディレクトリー名とファイル名

すべての Net Search Extender コマンドで、ディレクトリー名およびファイル名を SBCS 文字で指定しなければなりません。パス名 (ファイル名を含む) の最大長は 256 バイト以下です。

Outside-In ライブラリーのインストール

Net Search Extender を Stellent の Outside-In ソフトウェアと併用する場合、プラットフォームごとにライブラリーをセットアップする必要があります。

- Windows の場合、ライブラリーが置かれているディレクトリーを path 環境変数に追加してください。
- UNIX の場合、Outside-In ライブラリーをすべて /opt/IBM/db2/V8.1/lib ディレクトリーの中に追加してください。

インストール検査

Net Search Extender のインストール検査は、Windows および UNIX プラットフォームで使用できます。

注

分散データベースの場合、以下の検査サンプルを使用してください。

```
nsample_partitioned database_name [node_number][table_space_filename]
```

Windows でのインストール検査

Net Search Extender が正しくインストールされているか確認するには、次のステップを実行します。

- 次のステップに従って管理スクリプト `nseample.bat` を呼び出し、テキスト索引をセットアップします。
 1. `db2cmd` を呼び出して、DB2 コマンド・ウィンドウを開きます。
 2. `<sqllib>%samples%db2ext` に変更します。

3. DB2 コマンド・ウィンドウから `nsesample.bat <yourdb>` を呼び出します。ここで `<yourdb>` は、データベースの名前です。データベースがまだ存在していない場合、このコマンドを実行することで、そのデータベースが作成されることに注意してください。
 4. 現行ディレクトリーに生成された出力ファイル `nsesample.log` を確認します。
- その後、DB2 コマンド・ウィンドウから次のサンプル照会を呼び出し、実行します。
 1. `db2 connect to <yourdb>` を使用してデータベースと接続します。
 2. `db2 -tvf search` を使用してサンプル照会を実行します。
 3. スクリプト内に含まれる照会結果を確認します。どの照会も 1 つ以上のヒットを戻します。

`nsesample.log` ファイルにエラーがなく、すべての照会が機能していれば、Net Search Extender は正常にインストールされています。

UNIX でのインストール検査

Net Search Extender が正しくインストールされているか確認するには、次のステップを実行します。

- 次のステップに従って管理スクリプト `nsesample` を呼び出し、テキスト索引をセットアップします。
 1. `<instance_owner_home>/sqlllib/samples/db2ext` に変更します。
 2. `./nsesample <yourdb>` を呼び出します。データベースがまだ存在していない場合、このコマンドを実行することで、そのデータベースが作成されることに注意してください。
 3. ホスト・ディレクトリーに生成された出力ファイル `nsesample.log` を確認します。
- その後、同じ DB2 コマンド・ウィンドウからいくつかのサンプル照会を呼び出し、実行します。
 1. `db2 connect to <yourdb>` を使用してデータベースと接続します。
 2. `db2 -tvf search` を使用してサンプル照会を実行します。
 3. スクリプト内に含まれる照会結果を確認します。どの照会も 1 つ以上のヒットを戻します。

`nsesample.log` ファイルにエラーがなく、すべての照会が機能していれば、Net Search Extender は正常にインストールされています。

注

移行については、191 ページの『付録 A. 移行』を参照してください。

インストール検査

第 3 章 ユーザー・シナリオ

この章では、以下の順を追った例を使用して、Net Search Extender について学習します。

SQL スカラー検索関数の例

このコマンド行の例では、索引作成および使用可能な検索関数を示しています。

ストアード・プロシージャ検索の例

このコマンド行の例では、上記の例の索引コマンドを使用しています。しかしながら、この例では、キャッシュを追加することにより、使用可能な異なる索引作成関数と検索関数を示しています。

SQL 表値関数の例

Net Search Extender の使用について詳しくは、以下の章を参照してください。

- 23 ページの『第 4 章 計画』
- 31 ページの『第 6 章 テキスト索引の作成と保守』
- 73 ページの『第 8 章 検索』

注

例を使用する前に、インストール検査手順を使用して、Net Search Extender が正常にインストールされていることを確認してください。

SQL スカラー検索関数の場合の簡単な例

DB2 Net Search Extender の例では、以下のステップを使用します。

1. データベースを作成する
2. データベースをテキスト検索可能にする
3. 表を作成する
4. フルテキスト索引を作成する
5. サンプル・データをロードする
6. テキスト索引を同期させる
7. テキスト索引で検索する

既存データベースを使用して、オペレーティング・システムのコマンド行でサンプル・コマンドを発行することができます。以下の例では、データベース名を `sample` とします。

データベースを作成する

以下のコマンドを使用して、DB2 のデータベースを作成できます。

```
db2 "create database sample"
```

データベースをテキスト検索可能にする

オペレーティング・システムのコマンド行で、DB2 コマンドと同じように DB2 Net Search Extender コマンドを発行できます。たとえば、Net Search Extender インスタンス・サービスを開始するには、以下コマンドを使用します。

```
db2text "START"
```

DB2 Net Search Extender で使用するデータベースを作成するには、以下のコマンドを使用します。

```
db2text "ENABLE DATABASE FOR TEXT CONNECT TO sample"
```

このステップは、それぞれのデータベースで 1 回だけ実行します。

表を作成する

```
db2 "CREATE TABLE books (isbn VARCHAR(18) not null PRIMARY KEY,  
author VARCHAR(30), story LONG VARCHAR, year INTEGER)"
```

この DB2 コマンドによって、books という名前の表が作成されます。この表には、author、story、isbn 番号、および本が出版された year に関する列が含まれます。

フルテキスト索引を作成する

```
db2text "CREATE INDEX db2ext.myTextIndex FOR TEXT ON books (story)  
CONNECT TO sample"
```

このコマンドで、列 story のフルテキスト索引が作成されます。テキスト索引の名前は db2ext.myTextIndex です。

サンプル・データをロードする

```
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)"
```

これらのコマンドで、3 冊の本の isbn、author、story、および publishing year が表にロードされます。

テキスト索引を同期させる

sample 表のデータでテキスト索引を更新するには、以下のコマンドを使用します。

```
db2text "UPDATE INDEX db2ext.myTextIndex FOR TEXT CONNECT TO sample"
```

テキスト索引で検索する

テキスト索引を検索するには、以下の CONTAINS スカラー検索関数を使用します。

```
db2 "SELECT author, story FROM books WHERE CONTAINS  
(story, '¥"cat¥') = 1 AND YEAR >= 2000"
```

注

使用するオペレーティング・システム・シェルによっては、テキスト検索句を囲む二重引用符の前に、別のエスケープ文字が必要な場合があります。上記の例ではエスケープ文字として「¥」を使用しています。

この照会では、2000 年以降の、cat という語に関するすべての本が検索されます。照会によって、次の結果表が戻されます。

```
AUTHOR    STORY
Mike     The cat hunts some mice.
```

他に、SCORE および NUMBEROFMATCHES 関数がサポートされています。SCORE は、見つかった文書を検索引き数がどの程度正確に表現しているかを示す標識を戻します。NUMBEROFMATCHES は、結果の文書内にある照会語のマッチ回数を戻します。

キャッシュを使用したストアード・プロシージャ検索の場合の簡単な例

DB2 Net Search Extender ストアード・プロシージャ検索の例では、以下のステップを使用します。

1. キャッシュ・オプションを指定してテキスト索引を作成する。
2. 索引を同期させ、キャッシュをアクティブ化する。
3. TEXTSEARCH ストアード・プロシージャで検索する。

注

ストアード・プロシージャの例は、前の例が完了して、データベースがまだ使用可能であることを前提としています。

キャッシュ・オプションを指定してテキスト索引を作成する

データベースがすでに使用可能になっているため、以下のコマンドを使用してフルテキスト索引を作成します。

```
db2text "CREATE INDEX db2ext.mySTPTextIndex FOR TEXT ON books (story)
        CACHE TABLE (author, story) MAXIMUM CACHE SIZE 1
        CONNECT TO sample"
```

この例では、フルテキスト索引は、列 story に関するものであり、列 author および story のキャッシュ表が指定されています。テキスト索引の名前は mySTPTextIndex です。

索引を同期させ、キャッシュをアクティブ化する

表に挿入されたデータに基づいて索引を更新するには、以下のコマンドを使用します。

```
db2text "UPDATE INDEX db2ext.mySTPTextIndex FOR TEXT CONNECT TO sample"
```

キャッシュをアクティブ化するには、以下のコマンドを使用します。

```
db2text "ACTIVATE CACHE FOR INDEX db2ext.mySTPTextIndex FOR TEXT
        CONNECT TO sample"
```

TEXTSEARCH ストアド・プロシージャーで検索する

DB2 Net Search Extender ストアド・プロシージャーは、特定のケースでのみ使用できます。詳細は、6 ページの『ストアド・プロシージャー検索の使用』を参照してください。

```
db2 "call db2ext.textSearch
      ('¥"cat¥"', 'DB2EXT', 'MYSTPTTEXTINDEX', 0, 2, 0, 0, ?, ?)"
```

この照会では、cat に関するすべての本が検索されますが、最初の 2 つの結果のみを戻します。この場合、照会によって次の結果表が戻されます。

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

検索構文のその他のサンプルについては、sql1lib¥sample¥db2ext¥search ファイルを参照してください。

照会で使用される他のパラメーターの詳細については、80 ページの『ストアド・プロシージャー検索を使用するテキスト検索』を参照してください。

SQL 表値関数の場合の簡単な例

SQL 表値関数は、直前の例で作成されたテキスト索引に対して使用できます。

SQL 表値関数照会は、前に使用した CONTAINS 照会に対応しています。詳細は、20 ページの『テキスト索引を同期させる』を参照してください。

```
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
```

詳細は、7 ページの『検索での SQL 表値関数の使用』を参照してください。

注

CAST (NULL AS VARCHAR(18)) は、表 books の主キーに対して対応する表値関数を呼び出します。

それぞれの主キーのタイプごとに、1 つの表値関数があります。これによって、DB2 の正しい表値関数が識別されます。

第 4 章 計画

DB2 Net Search Extender を最も効果的に使用するには、開発の前になんらかのプランを作成することが重要です。この開発には、データベース管理者、インターフェースおよびシステム的设计者、開発者などの、いくつかのグループの参画が望まれます。

以下のセクションでは、考慮すべき領域について説明しています。

DB2 Net Search Extender ベースのアプリケーションの開発について詳しくは、以下の章を参照してください。

- 27 ページの『第 5 章 Net Search Extender インスタンス・サービス』
- 31 ページの『第 6 章 テキスト索引の作成と保守』
- 73 ページの『第 8 章 検索』

ディレクトリーのロケーションおよび索引ストレージ

索引に必要なディスク・スペースは、索引付けする必要のあるデータのサイズとタイプによって異なります。参考として、単一バイト文書に索引付けする場合は、索引付けする必要のある文書のサイズの 0.7 倍のディスク・スペースを確保する必要があります。2 バイト文書の場合は、索引付けする必要のある文書のサイズと同じディスク・スペースを確保してください。

作業ディレクトリー内の一時ファイルに必要なスペース量は、索引ディレクトリー内の最終的な索引ファイルに必要なスペース量の 1.0 から 4.0 倍です。

複数の大きな索引がある場合、特に索引の更新時または検索時に複数の索引に並行アクセスする場合は特に、各索引を別個のディスク装置に保管する必要があります。

テキスト索引の保管先となるディレクトリーも指定する必要があります。十分なディスク・スペースがあること、また、DB2 インスタンス所有者がディレクトリーに対する書き込みアクセスを持っていることを確認してください。

文書フォーマットおよびサポートされるコード・ページ

DB2 Net Search Extender は、検索対象のテキスト文書のフォーマット（またはタイプ）を知る必要があります。この情報は、テキスト文書の索引作成時に必要になります。

DB2 Net Search Extender は、以下の文書フォーマットをサポートします。

TEXT プレーン・テキスト（例：フラット ASCII）

HTML ハイパーテキスト・マークアップ言語

XML 拡張マークアップ言語

GPP 汎用パーサー・フォーマット (ユーザー定義タグ付きのフラット・テキスト)

Outside-In (INSO)

PDF や他の一般的なテキスト・フォーマット・ツール (Microsoft Word など) からテキスト内容を抽出するためのフィルター・ソフトウェア。詳細については『Outside-In フィルター・ソフトウェア』を参照してください。

文書フォーマット HTML、XML、GPP、および Outside-In フィルター・フォーマットの場合、検索は文書の特定部分に制限することができます。87 ページの『第 9 章 構造化文書を使用する作業』は、文書モデルを定義し、処理する方法について説明しています。

サポートされていない文書フォーマットに対して Outside-In フィルターが使用できない場合、ユーザー定義関数 (UDF) を作成できます。この UDF は索引作成時に指定する必要があり、これを使用して、サポートされていないフォーマットからサポートされているフォーマットにデータを変換します。

詳細については、120 ページの『CREATE INDEX』を参照してください。

サポートされているコード化文字セット ID (CCSID) の文書である場合、その文書は索引付けできます。CCSIDは、コード・ページとも呼ばれます。サポートされるコード・ページのリストは 209 ページの『付録 D. サポートされている CCSID』を参照してください。

データベースのコード・ページをチェックするには、次の DB2 コマンドを使用してください。

```
db2 GET DB CFG for <dbname>
```

整合性のために、DB2 は通常、文書のコード・ページをデータベースのコード・ページに変換します。ただし、DB2 データベース内のデータを、BLOB、FOR BIT DATA、または datalink value などのバイナリー・データ・タイプの列に保管する場合には、DB2 はデータを変換しないので、文書は元の CCSID を保持します。

2 つの異なるコード・ページを使用すると、テキスト索引の作成時または検索時に問題が発生する原因になる可能性があります。詳しくは、36 ページの『バイナリー・データ・タイプのテキスト索引の作成』を参照してください。

Outside-In フィルター・ソフトウェア

DB2 Net Search Extender は、サード・パーティーのフィルター・ソフトウェアをサポートします。Stellent の Outside-In というこのソフトウェアを使用することにより、PDF ファイルや、一般的なテキスト・フォーマット・ツールの独自のフォーマットで作成された文書から、ネイティブ・アプリケーションを使わずにテキスト内容を抽出することができます。フォーマットの例としては、Microsoft Word や Lotus WordPro などがあります。

Outside-In のライブラリーは、UPDATE INDEX の実行中にプラグインとしてロードできます。ライブラリーは Net Search Extender の一部ではないので、別途インストールする必要があります。Net Search Extender が Outside-In ライブラリーを見

つけられるようにする必要があります。ライブラリーのセットアップおよび使用について詳しくは、16 ページの『Outside-In ライブラリーのインストール』を参照してください。

Outside-In ソフトウェアは、テキスト内容だけでなく、フィールドなどの構造情報も生成することができます。Net Search Extender は、Outside-In が生成する文書情報のうち、どの部分を索引に保管するかをカスタマイズすることができます。これを行うには、特定のタイプの文書モデルである Outside-In 文書モデルを適用する必要があります。

詳しくは、184 ページの『Outside-In フィルター文書の場合の文書モデルの定義』を参照してください。

Stellent の Web サイトは <http://www.stellent.com> です。

フィルター・フォーマットのリストをご覧になるには、以下の URL をご使用ください。

```
http://www.stellent.com/intradoc-cgi/nph-idc_cgi.exe/p31019225.pdf?IdcService=GET_FILE&noSaveAs=1&Rendition=Web&RevisionSelectionMethod=LatestReleased&allowInterrupt=1&dDocName=p31019225
```

注

Outside-In フィルター・ソフトウェアは、32 ビット・インスタンスでのみ使用できます。64 ビットではサポートされません。

ユーザーの役割

Net Search Extender のユーザーには、以下のような、いろいろな役割および許可があります。

DB2 インスタンス所有者

DB2 インスタンス所有者は、DB2 Net Search Extender のインスタンス・サービスの開始と停止を実行でき、ロッキング・サービスをコントロールできます。さらに、DB2 インスタンス・ユーザーは、各使用可能データベースの DBADM になります。これにより、DB2 Net Search Extender によって行われるすべてのデータベース変更の集中制御が可能になります。

必須 DB2 権限

DBADM が ENABLE DATABASE に権限付与されます。

必須ファイル・システム権限

すべてのテキスト索引ディレクトリーの読み取りおよび書き込み、モデル・ファイルへの読み取りアクセス。

インスタンス所有者用のコマンド

DB2TEXT START、DB2TEXT STOP、DB2TEXT CONTROL。

これらのコマンドは、サーバーでだけ使用できます。分散 DB2 環境の場合は、どのサーバーでも使用できます。各コマンドは、コマンドを実行中のユーザーが DB2 インスタンス所有者であるかどうかをチェックします。

ユーザーの役割

UNIX システム上で個別の隔離ユーザー ID を使用しても、Net Search Extender 処理の権限またはパフォーマンスは影響を受けないことに注意してください。

データベース管理者

データベース管理者は、DB2 Net Search Extender で使用するデータベースを使用可能または使用不可にできます。

必須 DB2 権限

DBADM (ENABLE DATABASE の場合は SYSADM)。

データベース管理者用のコマンド

DB2TEXT ENABLE DATABASE、および DB2TEXT DISABLE DATABASE。

テキスト表所有者

テキスト表所有者は索引の作成、ドロップおよび変更を行えます。このユーザーは、索引のロケーションの制御、およびフルテキスト索引の更新の制御ができなければならないことに注意してください。

必須 DB2 権限および特権

テキスト表の所有者。

テキスト表所有者用のコマンド

DB2TEXT CREATE INDEX、DB2TEXT DROP INDEX、DB2TEXT ALTER INDEX、DB2TEXT ACTIVATE CACHE、DB2TEXT DEACTIVATE CACHE、DB2TEXT UPDATE INDEX、DB2TEXT CLEAR EVENTS、および DB2EXTTH。

DB2 インスタンス所有者のユーザー ID を使用した場合、コマンドの実行が不完全になることに注意してください。したがって、テキスト索引を作成または変更する前に、インスタンス所有者に必要なファイル・システム・アクセスを付与してください。

索引作成でのコマンド行または DB2 コントロール・センター・インターフェースの使用

索引の作成は、コマンド行を使用しても、DB2 コントロール・センターのインターフェースを使用しても行うことが可能です。

第 5 章 Net Search Extender インスタンス・サービス

DB2 Net Search Extender インスタンス・サービスは、次のサービスから構成されています。

- ロッキング・サービス
- 更新サービス

この章では、DB2 Net Search Extender インスタンス・サービスの開始と停止の方法について説明します。また、ロッキング・サービスと更新サービスについても詳しく説明します。

DB2 Net Search Extender の開始および停止

テキスト索引を作成し、文書を検索する前に、DB2 Net Search Extender インスタンス・サービスを開始する必要があります。

インスタンス・サービスを開始するには、DB2 インスタンスの所有者ユーザー ID (UNIX システムのみ) にログオンして、次のコマンドを入力します。

```
db2text start
```

インスタンス・サービスを停止するには、次のコマンドを入力します。

```
db2text stop
```

1 つの DB2 インスタンスあたりに 1 つの Net Search Extender インスタンス・サービスがなければならないことに注意してください。ロッキング・サービスは、複数のデータベースのロックを保持します。

ロッキング・サービス

DB2 Net Search Extender を開始すると、ロッキング・サービスは自動的に開始します。ロッキング・サービスは、Net Search Extender のテキスト索引への同時アクセスの同期をとるために必要です。

ロッキング・サービスは、2 つの処理で同時にテキスト索引を変更しようとしたり、別の処理でテキスト索引を変更中に、他の処理で同じテキスト索引データを読み取ったりすることがないようにします。したがって、大半の処理で開始前にテキスト索引に対するロックが要求され、処理が完了すると再びロックが解放されます。

Net Search Extender テキスト索引のロッキング・サービスと DB2 表へのアクセスを制御する DB2 ロックとを混同しないように注意してください。

ロッキング・サービスの使用

Net Search Extender には、索引への同時アクセスを制御する、さまざまなタイプのロックがあります。検索要求の場合のようにテキスト索引を読み取るだけなのか、

ロッキング・サービスの使用

それともテキスト索引への変更を計算してからファイルに書き込む必要があるのかといったことによって、異なるロックが使用されます。

db2text start 時に、ロッキング・サービスは自動的に開始します。テキスト索引に対する次のタイプのロックがあります。

S-lock 共有読み取り専用アクセス用。例：検索要求など。

U-lock

同時読み取りアクセスに対する索引への変更 (更新) を行っている間の読み取りおよび書き込みアクセス用。例：検索要求など。

X-lock 変更が実際に索引に書き込まれる間の短時間の排他的読み取り/書き込みアクセス用。

IX-lock

更新処理が X-lock を待機中に新規の S-locks を阻止するために意図された排他的読み取り/書き込みアクセス用。

DB2 インスタンスあたりに 1 つの Net Search Extender ロッキング・サービスが存在します。ロッキング・サービスは、複数のデータベースのロックを保持します。

ロッキング・サービス構成ファイルは、db2ext1m.cfg です。UNIX システムの場合は <instance_owner_home>/sql1lib/db2ext 上に保管され、Windows の場合は <sql1lib>%<DB2INSTANCE>%db2ext に保管されます。

構成ファイルの変更は、db2text start 時に Net Search Extender インスタンス・サービスが開始されて初めて有効となります。詳しくは、98 ページの『CONTROL』を参照してください。ユーザーは以下の値を設定できます。

- データベースの最大数
- データベースあたりの索引の最大数
- 索引あたりの許可されたロック (並行ユーザー) の最大数
- ロックを取得するための待機時間および試行回数

構成ファイルのデフォルト値は、次のとおりです。

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

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

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

  latchTimeout = "80"
/>
```

構文は <default attribute=value.../> で、属性とその意味は、次のとおりです。

maxDbs

ロッキング・サービスで処理できるデータベースの数 (integer 値 >1)。

maxIdxPerDb

ロックできるデータベースあたりの索引数 (integer 値 >1)。この値は、すべてのデータベースに対して同一です。

maxLocksPerIdx

索引上に同時に存在できるロックの数 (integer 値 >1)。この値は、すべての索引に対して同一です。

sWait/sAttempt

S-lock を要求する際に sAttempt は、ロックが即時に付与されない場合に試行される数です。 sWait はこれらの 2 つの試行の間の待ち時間 (integer 値 >1) です。これらのパラメーターは、IX-locks にも適用されます。

uWait/uAttempt

U-lock を要求する際に uAttempt は、ロックが即時に付与されない場合に試行される数です。 uWait はこれらの 2 つの試行の間の待ち時間 (integer 値 >1) です。

xWait/xAttempt

X-lock を要求する際に xAttempt は、ロックが即時に付与されない場合に試行される数です。 xWait はこれらの 2 つの試行の間の待ち時間 (integer 値 >1) です。

latchTimeout

これは、インターバル・ロッキング・サービスの追加の待ち時間です。ロックのための待ち時間の合計を決定するには、次の計算を使用します。

$$\text{waiting time} = \# \text{ attempts} * (\# \text{ waits} + (2 * \# \text{ latchTimeout}))$$

待ち時間は、ミリ秒単位で計算されます。試行のたびに latchTimeout 値は、待ち時間の合計に追加される際に 2 倍にされることに注意してください。

ロック・スナップショットの表示

次のいずれかのコマンドを使用すると、ロック・スナップショットを表示できます。

- 単一のテキスト索引の場合

```
db2text CONTROL LIST ALL LOCKS FOR DATABASE mydatabase INDEX myindex
```

- データベースの、ロックされたすべてのテキスト索引の場合

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

実際にロックされている索引だけがリストされることに注意してください。

初めてテキスト索引がロックされると、メモリーはデータベースおよびロッキング・サービスのテキスト索引の両方のために予約されます。さらにテキスト索引がロックされる場合は、メモリーはロッキング・サービスのこれらのテキスト索引にも割り振られます。このメモリーは、テキスト索引をドロップしたか、データベースを使用不可にした場合、あるいは Net Search Extender サービスを再始動した場合にだけ再び解放されます。つまり、現在、ロックが設定されていなくても、ロッキング・サービスでは、テキスト索引またはデータベースはメモリーを消費することになります。

ロック・スナップショットの表示

コマンド「db2text CONTROL CLEAR ALL LOCKS」は、データベースまたは索引上のすべてのロックを強制的に解放します。このコマンドの使用方法の詳細に関しては、98 ページの『CONTROL』を参照してください。このコマンドは、データベースまたは索引に割り振られたメモリーを解放しないことに注意してください。メモリーを解放するには、索引をドロップするか、データベースを使用不可にする、または Net Search Extender サービスを再始動する必要があります。

更新サービス

更新サービスは db2text start の時に開始し、指定した時間にテキスト索引を自動的に更新します。テキスト索引は、ユーザー表と即時に同期がとられるわけではないので、注意してください。

索引の作成の際に、次のコマンドを使用することにより、更新サービスが、索引の更新が必要かどうかをどの程度の頻度でチェックするかを指定できます。

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

この例では、毎週の月曜日と水曜日の午後 12 時と午前 12 時に更新サービスが起動し、索引 db2ext.title に実行するべき作業が存在するかどうかをチェックします。少なくとも 5 回の変更があって初めて、自動索引更新とデータベースとの同期化を開始することができます。

パラメーターに関する詳細は、120 ページの『CREATE INDEX』を参照してください。

パーティション・データベース環境の場合、更新サービスは 1 つのノードでしか開始しません。

注

UPDATE FREQUENCY を毎分にすると、マシンのワークロードが高くなります。これを回避するには、非常に限定された数の索引に対してだけ UPDATE FREQUENCY を使用するようしてください。

第 6 章 テキスト索引の作成と保守

この章では、テキスト索引の作成と保守について説明します。以下の項目をとりあげます。

- db2text コマンドの紹介
- データベースをテキスト検索可能にする
- さまざまなデータ・タイプのテキスト索引の作成
- DB2 レプリケーションで、インクリメンタル索引更新による、ニックネームのテキスト索引を作成する
- ストアード・プロシージャで利用できるテキスト索引の作成
- ビューに基づくテキスト索引
- 索引の保守

この章では、発生する可能性があるコード・ページの問題の回避、および考慮する必要があるパフォーマンスについても説明します。

テキスト索引を作成する前に、23 ページの『第 4 章 計画』にある前提条件を確認してください。その他の索引作成の前提要件としては、db2text start コマンドを使用して、DB2 Net Search Extender インスタンス・サービスを開始することが含まれます。

テキスト索引の作成およびテキストの検索を可能にする例については、19 ページの『第 3 章 ユーザー・シナリオ』を参照してください。

注

DB2 コントロール・センターを使用して、テキスト索引を作成し、保守することもできます。51 ページの『第 7 章 DB2 コントロール・センターの使用』を参照してください。

db2text コマンドの紹介

DB2 Net Search Extender コマンドの例を次に示します。

```
db2text ENABLE DATABASE FOR TEXT
```

注

db2text ENABLE DATABASE FOR TEXT および db2text CREATE INDEX などの db2text コマンドもコマンドと呼びます。

ヒント

作成と索引保守のすべてのコマンドで、データベース、ユーザー、およびパスワードを指定できます。

```
db2text ... connect TO <database> USER <userID> USING <password>
```

db2text コマンドで接続オプションを省略すると、環境変数 DB2DBDFT によってデータベースが指定されることに注意してください。

コマンドのリストを表示するには、次のコマンドを入力します。

```
db2text ?
```

各コマンドの構文を表示するには、次のコマンドを入力します。

```
db2text ? command
```

たとえば、CREATE INDEX コマンドの構文を表示するには、次のコマンドを使用します。

```
db2text ? CREATE INDEX
```

db2text は、コマンドが正常に処理された場合 0 を、コマンドが処理されなかった場合は 1 を戻します。文書エラーがあるにもかかわらず、索引が更新される場合、db2text コマンドは 0 を戻します。

注

システム・シェルは、?、(、)、*、!、および " などの特殊文字を解釈します。したがって、コマンドにこれらの文字が含まれている場合には、引用符またはエスケープ文字を使用してください。

以下に、特殊文字を使用する UNIX コマンドの例を示します。

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

データベースを使用可能にする

サマリー

時期	検索対象のテキストの列を含んでいるデータベースごとに 1 回。
コマンド	ENABLE DATABASE FOR TEXT
権限	SYSADM

このコマンドは、接続されたデータベースを DB2 Net Search Extender が使用できるように準備します。

また、このコマンドは、155 ページの『第 15 章 SQL スカラー検索関数および SQL 表値関数』で説明されている DB2 Net Search Extender の検索関数および検索プロシージャを宣言します。

データベースを使用可能にすると、次の表およびさまざまなビューが作成されます。

db2ext.dbdefaults

索引、テキスト、および処理特性のデータベース・デフォルト値を示します。

db2ext.textindexformats

サポートされるフォーマットおよび使用されるモデル・ファイルのリストを示します。

db2ext.indexconfiguration

索引構成パラメーターを示します。

db2ext.textindexes

すべてのテキスト索引を追跡するカタログ・ビュー。

db2ext.proxyinformation

プロキシ・サーバーを使用してファイルをアクセスするためのプロキシ情報を示します。

すべてのビューについては、199 ページの『付録 C. Net Search Extender インフォメーション・カタログ』を参照してください。

いったんデータベースが使用可能になると、使用不可にするまでは使用可能の状態が続きます。

注

インスタンス所有者 ID と異なる fenced ユーザー ID では、パーティション・データベースを扱う作業ができません。

データベースを使用不可にする

サマリー

時期 このデータベースでテキスト検索を行う必要がなくなった場合。

コマンド

DISABLE DATABASE FOR TEXT

権限 データベースに対する DBADM

DB2 Net Search Extender がデータベースを使用できるように準備するとき、管理上の変更を行う場合があります。このセクションでは、このプロセスを元に戻す場合に役立つ関数について説明します。

データベースを使用不可にする

接続データベースを使用不可にするには、次のコマンドを使用します。

```
db2text DISABLE DATABASE FOR TEXT
```

データベースを使用不可にすると、以下のオブジェクトも削除されます。

- データベースを使用可能にした時に作成された DB2 Net Search Extender カタログ・ビュー。
- DB2 Net Search Extender の SQL 関数 (UDF) の宣言。

データベースを使用不可にし、すべてのテキスト索引を除去するには、次のコマンドを使用します。

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

注

データベースにテキスト索引が定義されている場合に、そのデータベースを使用不可にしようとする、失敗します。これらの索引を 1 つ 1 つ除去した上で、何か問題が生じるかどうかを検査するようお勧めします。 `disable database for text force` コマンドを使用した場合に保証されるのは、データベース内の Net Search Extender カタログ表が削除されるということに過ぎません。

しかし、完全にドロップできない索引があった場合は、手作業でクリーンアップしなければならないリソースがまだあるかもしれません。以下のものが含まれます。

- 索引ディレクトリー、作業ディレクトリー、およびキャッシュ・ディレクトリー内のファイル
- `ctedem.dat` 内のスケジューラー項目
- レプリケーション・キャプチャー・オプションを使用して索引を作成した場合には、リモート・データベースの表にある、`IBMSNAP_SIGNAL`、`IBMSNAP_PRUNE_SET`、および `IBMSNAP_PRUNCNTL` の各項目を手作業で削除しなければなりません。これらの項目は、
`APPLY_QUAL="NSE" || <instance name> and TARGET_SERVER= <database name>`
コマンドで簡単に識別できます。

次の例では、インスタンスは DB2、データベースは SAMPLE です。

```
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';
```

テキスト索引の作成

サマリー

時期	検索対象のテキストを含んでいる列ごとに 1 回。
コマンド	CREATE INDEX ... FOR TEXT ... (以下の例を参照)
権限	表に対する CONTROL

サポートされているデータ・タイプのテキスト索引を作成できます。ただし、以下のデータ・タイプでは要件が異なります。

- バイナリー・データ・タイプ
- サポートされていないデータ・タイプ
- DATALINK データ・タイプ

また、ストアド・プロシージャ用のテキスト索引の作成に関しても要件は異なります。

テキスト索引を作成すると、以下のオブジェクトも作成されます。

ログ表 ログ表は、ユーザー表内で変更されたすべての行を追跡します。 **Recreate index on Update** オプションを選択する場合、またはレプリケーション・キャプチャー表を使用する場合には、ログ表は作成されないので注意してください。

イベント表

この表は、テキスト索引の更新時の問題に関する情報を収集します。

ユーザー表のトリガー

トリガーによって、列内の文書が追加、削除、または変更された場合は常に、ログ表に情報が追加されます。この情報は、次の索引作成時に索引を同期化するときのために必要です。

トリガーが作成されるのは、ログ表を作成し、かつ基本表 (ビューでもニックネーム表でもない) のテキスト索引を作成した場合だけであることに注意してください。

パフォーマンスおよびディスク・スペースを最適化するために、CREATE INDEX コマンドを使用して、表のために別の表スペースを指定できます。

注

DB2 LOAD コマンドを使用して文書をインポートすると、問題が発生する可能性があります。これは、トリガーが開始されず、ロードされた文書の索引を増大して作成することができないためです。

したがって、DB2 IMPORT コマンドを使用することをお勧めします。このコマンドはトリガーをアクティブ化します。

テキスト索引の作成

次の例では、表 `htmltab` 内のテキスト列 `HTMLFILE` のテキスト索引を作成します。

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

この表の主キーが必要です。

索引作成のデフォルト値は、`db2ext.dbdefaults` ビューから取られます。

索引作成時にエラーが発生した場合には、いわゆる**索引更新イベント**がイベント表に追加されます。たとえば、索引作成のためにキューに入れられた文書が見つからない場合などです。詳細については、205 ページの『イベント・ビュー』を参照してください。

`CREATE INDEX` が行った変更を元に戻すには、`DROP INDEX` コマンドを使用します。詳細については、49 ページの『テキスト索引のドロップ』を参照してください。

テキスト索引とデータベースを同期させるには、次のコマンドを使用します。

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

同期が終了するまで、文書の検索を行うことはできません。

検索のサマリー

索引作成中に選択するオプションによって、以下のようなさまざまな検索を行うことができます。

- **SQL スカラー検索関数**は、ビューに関して作成されたテキスト索引以外のすべてのテキスト索引を処理します。
- **ストアード・プロシージャ検索関数**は、キャッシュを使用して作成されたテキスト索引のみを処理します。
- **SQL 表値関数**は、ビューに作成されたテキスト索引を含むすべてのテキスト索引を処理します。

バイナリー・データ・タイプのテキスト索引の作成

`BLOB`、`FOR BIT DATA`、または `datalink` 値などのバイナリー・データ・タイプの列にデータを保管する場合、`DB2` はデータを変換しません。これは、文書が元のコード・ページ (`CCSID`) を保持することを意味します。これによって、テキスト索引を作成するときに問題が生じる可能性があります。これは、2 つの異なるコード・ページが存在することになるためです。よって、データベースのコード・ページを使用するか、`CREATE INDEX` コマンドで指定したコード・ページを使用するかを決定する必要があります。

この問題を回避するために、テキスト索引の作成時にコード・ページを指定してください。

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

コード・ページを指定しなかった場合には、次のコマンドを実行して、索引を作成するために使用された `CCSID` を調べてください。

```
db2 SELECT ccsid FROM db2ext.textindexes WHERE indname = 'COMMENT'
      and indschema = 'DB2EXT'
```

コード・ページが異なる複数の文書を 1 つのテキスト索引に含めることはサポートされていないことに注意してください。DB2 が文書のコード・ページの設定値を変換する方法については、「*DB2 Universal Database 管理ガイド*」を参照してください。viii ページの『関連情報』を参照してください。

文字データ・タイプの索引作成に関しては、このような問題がないことに注意してください。

サポートされないデータ・タイプのテキスト索引の作成

索引を作成するには、テキスト列が以下のいずれかのデータ・タイプでなければなりません。

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

文書がユーザー定義タイプ (UDT) などの上記以外のタイプの列にある場合には、入力としてユーザー・タイプを受け取り、出力タイプとして上記のいずれかのタイプを戻す関数を用意する必要があります。

この変換関数の名前を指定します。詳しくは、120 ページの『CREATE INDEX』を参照してください。

例：圧縮テキストを表に保管することにします。

1. 対話式 SQL セッションで、テキストのユーザー定義タイプ (UDT) を作成します。

```
db2 "CREATE DISTINCT TYPE COMPRESSED_TEXT AS CLOB(1M)"
```

2. 表を作成し、テキストを表に挿入します。

```
db2 "CREATE TABLE UDTTABLE (author VARCHAR(50) not null,
                             text COMPRESSED_TEXT, primary key (author))"
db2 "INSERT ..."
```

3. 仮に `uncompress` という名前でユーザー定義関数 (UDF) を作成します。これは、タイプ `COMPRESSED_TEXT` の値を受け取り、対応する圧縮解除したテキストを、たとえば `CLOB(10M)` 値として戻します。

4. 以下のようにテキスト索引を作成して、`uncompress` UDF を指定します。

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

DATALINK データ・タイプのテキスト索引の作成

DB2 Net Search Extender は、データ・タイプ DATALINK をサポートします。

1. プロキシを使用する場合は、db2ext.proxyinformation 表に 1 行を追加し、ホスト名、タイムアウトの秒数、使用されるポートが 'proxy' か 'socks' かを登録します。

```
db2 INSERT into db2ext.proxyinformation values
('hostname', '80' 'proxy', 10)
```

この表には 1 行のみが許可されます。これは、トリガーによって保証されません。

2. 異なるプラットフォーム上に Java 環境をセットアップする方法の詳細については、DB2 インフォメーション・センターなどにある DB2 Universal Database バージョン 8 の資料を参照してください。基本的に、データベース・マネージャ構成を調整しなければなりません。

データ・リンク UDF は、BLOB (100 KB) を戻します。戻りサイズを変更するには、DB2EXTDL を使用して、データベース・マネージャ構成パラメーター java_heap_sz を更新します。

データ・リンク jar ファイルのインストール

Net Search Extender を使用すると、DB2 データ・リンク機能によって参照されているファイルの中にあるデータに対して、索引を付けることができます。このためには、データ・リンク jar ファイル ctedludf.jar をインストールする必要があります。db2 コマンド行プロセッサ内で、次のコマンドを実行します。

- UNIX の場合:

```
call sqlj.install_jar
('file:/<instance_owner_home>/sqllib/java/ctedludf.jar','ctedludf.jar')
```

- Windows の場合:

```
call sqlj.install_jar
('file:D:¥sqllib¥java¥ctedludf.jar','ctedludf.jar')
```

次のコマンドを使用して、Java ヒープ・サイズを更新する必要もあります。

```
db2 update dbm cfg using JAVA_HEAP_SZ 2048
```

インストールした jar ファイルを登録抹消するにはすべてのプラットフォーム上で共通に次のコマンドを使用します。

```
call sqlj.remove_jar('ctedludf.jar')
```

エラー・メッセージのリストについては、259 ページの『付録 K. データ・リンク・メッセージ』を参照してください。

DB2 レプリケーションで、インクリメンタル索引更新による、ニックネームのテキスト索引を作成する

レプリケーション・キャプチャ表を使用してニックネームのテキスト索引を作成するには、事前に以下のステップを実行しなければなりません。

レプリケーションを使用してニックネームのテキスト索引を作成する

注

以下のステップはプロセスの概要を示しているに過ぎず、例ではありません。

1. すべてのサーバー定義およびラッパー定義とともに、DB2 フェデレーテッド・データベースをセットアップします。
2. リモート・サーバーで、レプリケーション・コントロール表およびキャプチャー・プログラムをセットアップします。ニックネームのソース表が存在するのはここです。「DB2 レプリケーションのガイドおよびリファレンス バージョン 8」の第 2 章『レプリケーションの設定』を参照してください。DB2 がニックネームを自動的に作成しない場合は、以下の表に関する 1 つのスキーマ名を使用して、フェデレーテッド DB2 データベースにニックネームを作成しなければなりません。
 - IBMSNAP_SIGNAL
 - IBMSNAP_PRUNE_SET
 - IBMSNAP_PRUNCNTL
 - IBMSNAP_REGISTER
 - IBMSNAP_REG_SYNC (DB2 以外のリモート・ソースのみ)

このステップを完了すると、レプリケーション・コントロール表のニックネームを、フェデレーテッド DB2 データベース上の 1 つの「キャプチャー・コントロール・スキーマ」の下にニックネームとして使用できるようになります。このスキーマ名は、DB2TEXT CREATE INDEX コマンドにとって重要です。

3. 表をレプリケーション・ソースとして登録します。詳しくは、「DB2 レプリケーションのガイドおよびリファレンス バージョン 8」の第 3 章『表およびビューをレプリケーション・ソースとして登録する』を参照してください。索引を作成するニックネームを登録する際の制約事項については、126 ページを参照してください。
4. 登録ステップで DB2 によってニックネームが自動的に作成されない場合は、フェデレーテッド・データベースのレプリケーション・キャプチャー表のためにニックネームを作成してください。レプリケーション・キャプチャー表は、変更データ (CD) 表か整合変更データ (CCD) 表のいずれかです。このニックネームは、DB2TEXT CREATE INDEX コマンドのパラメーターです。

列名

IBMSNAP_OPERATION、IBMSNAP_COMMITSEQ、IBMSNAP_INTENTSEQ、および主キー列の名前は変更できません。

5. DB2 レプリケーション・ソースを使用する場合は、キャプチャー・プログラムが実行中であることを確認してください。キャプチャー・プログラムではコールド・スタートを使用しないよう強くお勧めします。コールド・スタートを使用する場合は、APPLY_QUAL LIKE 'NSE%' に関する IBMSNAP_SIGNAL 表内のすべての行を再び挿入する必要があります。以下の SQL ステートメントで、これがどのように行われるかを調べることができます。

レプリケーションを使用してニックネームのテキスト索引を作成する

```
INSERT INTO <capture control schema>.IBMSNAP_SIGNAL
SELECT CURRENT TIMESTAMP, 'CMD', 'CAPSTART', MAP_ID, 'P'
FROM <capture control schema>.IBMSNAP_PRUNCNTL
WHERE APPLY_QUAL LIKE 'NSE%';
```

- レプリケーションを使用してニックネームに対してテキスト索引を作成するには、以下の例を使用します。

```
DB2TEXT
CREATE INDEX <indexname> FOR TEXT ON <nickname> (< text column>)
REPLICATION CAPTURE TABLE <capture nickname>
CONTROL TABLE SCHEMA <capture control schema>
```

ストアード・プロシージャ検索で利用できるテキスト索引の作成

ストアード・プロシージャ検索を使用するには、CREATE INDEX コマンドの実行時にキャッシュ・オプションを指定する必要があります。これにより、指定したデータがメイン・メモリーに移動され、ハイパフォーマンスを実現できます。

ただし、検索のための最初の索引の更新の前に、文書の入っていない表の索引の更新を行ってしまうことがないように、表の中に文書があるか確認してください。こうすることで、索引作成のパフォーマンスは向上し、なおかつキャッシュ・メモリー要件をしっかりと予測することもできます。

ストアード・プロシージャ検索を使用すると、文書に関連した定義済みデータを迅速に戻すことができます。これを定義するには、CREATE INDEX コマンドで、キャッシュ表オプションを使用します。次に、ACTIVATE CACHE コマンドによって、指定したデータをメモリー・キャッシュに移動します。

注

このテキスト索引は、ビューに対して作成されたものでない限り、SQL スカラー検索関数でも使用できます。

分散 DB2 環境では、ストアード・プロシージャのために単一ノード上の管理表の表スペースを明示的に指定し、明示的にこのノードに接続しなければなりません。

適切なノードと接続するためには、DB2NODE 環境変数を使用します。

ストアード・プロシージャ検索用のテキスト索引を作成する際には、次のパラメーターを決定し、計算しなければなりません。

- キャッシュのタイプ。
- 索引の更新方法。
- Net Search Extender で使用できる最大メモリー量 MAXIMUM CACHE SIZE。
- 以後の文書更新に必要な空きメモリー量 PCTFREE。これは、インクリメンタル更新の場合のみです。

以下のキャッシュのタイプを使用できます。

一時キャッシュ

一時キャッシュは、DB2TEXT ACTIVATE CACHE コマンドの実行ごとに再作成され、DB2 表からメモリーへのデータのロードが必要となります。これ

ストアド・プロシージャ検索で使用できるキャッシュ・テキスト索引の作成

は、永続的キャッシュのアクティブ化より長い時間を取ります。特に大きな索引の場合には顕著です。しかし、検索のパフォーマンスは若干向上するかもしれません。

永続的キャッシュ

永続的キャッシュは、ディスク上に保守されていて、各 DB2TEXT ACTIVATE CACHE コマンドでオペレーティング・システムにより素早くメモリーにマップされます。インクリメンタル索引更新のシナリオでは、永続的キャッシュを索引とキャッシュ間の同期を認めるようにアクティブ化する必要があります。そうしない場合、次の DB2TEXT ACTIVATE CACHE コマンドがキャッシュを最初から再作成します。

テキスト索引の更新は、次の方法で行えます。

更新のオプションに索引の再作成を指定しない方法

削除した文書のスロットはキャッシュで再利用できないので、表への文書の削除および再挿入は避けてください。したがって、アクティブ化された索引では、キー列を変更しないようにする必要があります。

これは、インクリメンタル更新とも呼ばれます。

更新のオプションに索引の再作成を指定する方法

これは、更新するたびに索引を再作成する方法です。可能な場合は、キャッシュ列の式で可変長データ・タイプを使用してください。これにより、キャッシュ・スペースが節約できます。また CACHE TABLE 文節には、対応するキャスト式を使用してください。

最初の索引のアクティブ化以後に、文書の 50% 以上を挿入する場合は、このオプションを使用してください。

Net Search Extender には、CREATE INDEX メモリー・パラメーターを決定するのに役立つ 2 つの SQL 関数があります。それは、MAXIMUM CACHE SIZE 関数と PCTFREE 関数です。

インクリメンタル更新および再作成更新

以下の UDF 関数は、推奨される MAXIMUM CACHE SIZE 値をメガバイト (MB) で戻します。

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

以下のコマンドは、表の平均の行の長さのパラメーターを戻します。

```
SELECT AVG(LENGTH(cache column_1) + ... + LENGTH(cache column_n))
```

平均は、さらに文書を表に挿入するときに、大きく変わる場合があることに注意してください。キャッシュ列の数は、DB2TEXT CREATE INDEX コマンドの CACHE TABLE 文節で使用した列式の数に関連しています。

詳細に関しては、195 ページの『付録 B. 大量のメモリーの使用』を参照してください。

インクリメンタル更新のみ

以下の UDF 関数は、実際の文書数、および最大文書数に基づく推奨 PCTFREE 値を戻します。

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

ストアド・プロシージャ検索で使用できるキャッシュ・テキスト索引の作成

文書の実際数とは、メモリー・キャッシュを作成する最初の ACTIVATE CACHE コマンド実行時の、ユーザーの表の行数に相当します。

最大文書数とは、次の DB2TEXT ACTIVATE コマンド (一時キャッシュの場合)、または DB2TEXT ACTIVATE CACHE RECREATE コマンド (永続的キャッシュの場合) が実行される前の、ユーザーの表における文書の最大予想数のことです。

更新のたびに索引を再作成する場合は、PCTFREE を 0 に設定してください。

例

表に 10 000 行あるとします。そして、20 000 行を超えることはないと予想しているものとします。次の呼び出しを使用して、必要な PCTFREE 値を計算します。

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

最大行サイズが 20 000 で、キャッシュには 2 列あり、その平均サイズが 76 であるとして、次の呼び出しを使用して、サイズを戻します。

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

適切なパラメーターを決定した後、次の呼び出しを使用して、索引およびキャッシュ表を作成できます。

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

この例では、結果表を高速で戻すためにメイン・メモリーを使用して、索引の他に docid 列が作成されます。キャッシュ・メモリーの 10 % は、今後の文書のために予約され、キャッシュの最大は、5 MB に制限されています。

テキスト索引の更新

この索引で検索するには、索引を更新してから、アクティブ化する必要があります。これにより、指定した表キャッシュ式がデータベースからメモリーにコピーされます。

ACTIVATE または UPDATE の操作中に MAXIMUM CACHE SIZE に達した場合は、以下の処置を実行することをお勧めします。

- 次の順序で DB2TEXT コマンドを使用して、キャッシュを再作成してください。
DEACTIVATE CACHE、ALTER INDEX、MAXIMUM CACHE SIZE、および ACTIVATE CACHE RECREATE。

文書の頻繁な更新が予想される場合は、CACHE TABLE 文節のキャッシュ列式に固定サイズ・データ・タイプを使用することを考慮してください。以下の例は、更新操作時に同じキャッシュ記憶機構を使用する方法を示しています。

```
CACHE TABLE(cast(C1 as char(20)), cast(substr(C2,1,10) as char(10))....
```

この場合、必ず非変数データ・タイプのみを使用してください。

テキスト索引のキャッシュのアクティブ化および非アクティブ化

キャッシュをアクティブ化する前に、不正確な PCTFREE 計算が行われないようにするために、ペンディングになっているインクリメンタル更新を実行します。

テキスト索引をアクティブ化するには、以下のコマンドを使用します。

```
db2text ACTIVATE CACHE FOR INDEX db2ext.comment FOR TEXT
```

このコマンドによって、指定したキャッシュ表データがデータベースから取り出され、メモリーに保管されます。必要な時間は表のサイズによって異なります。

注

索引がアクティブ化されているときに `update index` を呼び出すと、キャッシュ表も更新されます。削除した文書はキャッシュ・メモリーのスロットを消費する必要があるため、PCTFREE を十分大きな値に設定してください。

ACTIVATE CACHE 呼び出しは、システムが停止されると、毎回再実行する必要があります。永続キャッシュを使用すると、新たなアクティブ化を素早く行うことができます。

注

永続キャッシュをアクティブ化していないときに更新を行うと、永続キャッシュはドロップされ、アクティブ化呼び出しのときに再作成されます。

リソースを節約するために、現在、必要でないすべての索引を非アクティブ化する必要もあります。

残りメモリーをチェックするには、次の呼び出しを使用します。

```
db2text control show cache status for database cte index db2ext.comment
```

この呼び出しは、索引がアクティブ化されているかどうか、指定したキャッシュ・スペースのうちのどのくらいの量が残っているかを表示します。

ビューに基づくテキスト索引

ビューに基づくテキスト索引は、ストアド・プロシージャや表値関数を使用して作成できますが、スカラー関数 (たとえば CONTAINS) を含めることはできません。

もう 1 つの大きな欠点として、ビュー上でトリガーを作成できないため、基礎を成す基本表での変更を認識できないという点があります。

したがって、インクリメンタル更新の場合、ユーザーは、テキスト索引とデータベースを同期化するために、追加、更新または削除を行った文書を知っている必要があります。このためには、すべての変更をログ表に追加する必要があります。この処理を以下のサンプルで示します。

1. 基本表を作成するために、以下のコマンドを使用します。

ビューに基づくテキスト索引

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

2. いくつかの項目を追加するために、以下のコマンドを使用します。

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

3. ビューを作成するために、以下のコマンドを使用します。

```
db2 "create view sampleview as select key, comment from sample"
```

4. テキスト索引を作成、更新、およびアクティブ化するために、以下のコマンドを使用します。

```
db2text "create index indexview for text on hde.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"
```

注

ビューに基づくテキスト索引を作成するには、キャッシュ表を指定する必要があります。正しいログ表を作成するには、ビュー上の索引に関するキー列を指定しなければなりません。この方法で索引を作成する場合、表値関数を含む索引も使用できます。

分散データベース環境でストアード・プロシージャ検索を使用する場合は、単一ノード上の管理表の表スペースを明示的に指定し、明示的にこのノードに接続しなければなりません。適切なノードと接続するためには、DB2NODE 環境変数を使用します。

5. 表を更新するために、以下のコマンドを使用します。

```
db2 "insert into sample values(4,'Bernhard','is working in the same floor  
      as Manja, but not as Claus')"  
db2 "insert into sample values(5,'Guenther','shares the office with Raiko')"
```

6. 次にログ表を更新します。ログ表の名前を取得するには、以下のコマンドを使用します。

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

以下にログ表のレイアウトを示します。

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

ログ表に項目を追加するには、以下のコマンドを使用します。

```
db2 "insert into sample values(0,CURRENT TIMESTAMP,4)"  
db2 "insert into sample values(0,CURRENT TIMESTAMP,5)"
```

最初の値は、操作 (0 = 挿入、1 = 更新、2 = 削除) を示します。2 番目の値は、常に CURRENT TIMESTAMP で、最後の値は挿入されたキーです。

7. 索引を再び更新するために、以下のコマンドを使用します。

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

これで、ストアード・プロシージャを使用して新しい値を検索できるようになりました。

パフォーマンスに関する考慮事項

索引作成時のパフォーマンスを向上するために、次の点について考慮してください。

- VARCHAR データ・タイプを使用して、LONG VARCHAR または CLOB の代わりに、テキスト文書を保管する。
- テキスト索引とデータベース・ファイルを保管するために、別個のハード・ディスクを使用する。
- VARCHAR タイプではなく、TIMESTAMP や INTEGER などの小さな主キー列を使用する。
- ユーザーのシステムに、このデータすべてに使用できる十分な実メモリがあることを確認する。メモリが不十分な場合、オペレーティング・システムは代わりにページング・スペースを使用します。これにより、検索パフォーマンスは低下します。

異なるプラットフォームのメモリ所要量の構成については、195 ページの『付録 B. 大量のメモリの使用』を参照してください。

- 索引作成の自動更新時または手動更新時に使用される更新コミット・カウント・パラメーターを使用すると、インクリメンタル索引作成時の索引作成パフォーマンスが低下します。このパラメーターは、初期更新プロセス時には使用されません。

COMMITCOUNT を設定しない場合、db2ext.textindexes からの NUMBER_DOCS パラメーターは更新されません。したがって、更新処理中に文書の数を表示するには、CONTROL LIST コマンドを使用してください。詳細については、98 ページの『CONTROL』を参照してください。

- エラー・メッセージと警告メッセージがイベント・ログ表に書き込まれる場合は、索引更新中のパフォーマンスが低下することがあります。

注

最新のパフォーマンスのヒントについては、DB2 Net Search Extender の Web サイト

www.ibm.com/software/data/db2/extenders/netsearch/index.html を参照してください。

テキスト索引の保守

この章では、テキスト索引を保守する方法、およびテキスト索引に関する有益な情報を得る方法について説明します。保守タスクは以下のとおりです。

1. テキスト索引の更新と再編成
2. テキスト索引の変更
3. 索引更新イベントの削除
4. テキスト索引のドロップ
5. 索引状況の表示

上記のタスクをいつでも、どのような順序でも実行できます。さらに、索引と使用可能データベースをバックアップおよびリストアする方法についても説明します。

テキスト索引、ディレクトリー名、および更新の情報を表示するコマンドについては、199ページの『付録 C. Net Search Extender インフォメーション・カタログ』を参照してください。

DB2 コントロール・センターでも、テキスト索引を保守できます。66ページの『テキスト索引の保守』を参照してください。

テキスト索引の更新と再編成

テキスト索引を最初に作成し、更新した後は、テキスト索引を最新のものに保持する必要があります。たとえば、テキスト文書をデータベースに追加する際、またはデータベース内の既存の文書を変更する際は、索引の内容とデータベースの内容の同期をとるために、文書に索引を付ける必要があります。同様に、データベースからテキスト文書を削除する場合、その用語を索引から除去する必要があります。

RECREATE INDEX ON UPDATE オプションなしでテキスト索引が作成された場合、トリガーは自動的に、新規文書、変更文書、および削除文書に関する情報を内部ログ表に保管します。よって、次回索引が更新される時は、ログ表で参照される文書の索引が作成されます。ビューに基づくテキスト索引についての詳細は、43ページの『ビューに基づくテキスト索引』を参照してください。

CREATE INDEX コマンドで RECREATE オプションを指定すると、更新ごとに索引全体が再作成されます。このオプションは、ログ表もトリガーも作成しません。

一般に、索引はときどき更新されます。既存の索引の更新頻度は、ALTER INDEX コマンドを使用して変更できます。

索引更新の頻度は、更新を行う時期、およびキューに入れられる、テキスト変更の最小数に関して指定します。指定した日時に、ログ表内の変更の数が指定した数より少なければ、索引は更新されません。

定期的な索引作成を注意深く計画してください。テキスト文書の索引作成は、多くの時間とリソースが必要なタスクです。必要な時間は、多くの要因によって変わります。たとえば、文書のサイズ、最後の索引更新以後に追加または変更したテキスト文書の数、およびプロセッサの能力などです。

注

次のような場合、DB2 表でロールバックおよびデッドロック状態が起こり得ます。

- 更新頻度が高い場合
- 変更トランザクションが頻繁に起こる場合
- トランザクションが長い場合

UPDATE INDEX コマンドを使用すれば、索引を要求時に即時に更新できます。

サマリー

時期 定期的な索引作成の実行を待たないで、索引を即時に更新する時。

コマンド

UPDATE INDEX

権限 表に対する CONTROL

次のコマンドは、索引を更新します。

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

このコマンドは、複数のテキスト文書をデータベースに追加済みであり、これらの文書を直ちに検索する必要がある場合に便利です。

手動の再編成が必要かどうかを判別するには、次のコマンドを使用して、db2ext.textindexes ビューを照会してください。

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

MANUAL REORGANIZATION を指定して列をしばしば更新する場合、更新処理は遅くなります。手動で再編成するには、次のコマンドを使用してください。

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

ただし、CREATE INDEX 時に AUTOMATIC REORGANIZE を指定する場合には、必要なときに索引は自動的に再編成されます。

テキスト索引の変更

サマリー

時期 更新頻度、または索引ディレクトリーと作業ディレクトリーを変更する必要がある時。

コマンド

ALTER INDEX

権限 表に対する CONTROL

このコマンドを使用して、索引の作業ディレクトリー、索引の更新頻度、またはキャッシュ特性、特に MAXIMUM CACHE SIZE または PCTFREE を変更します。更新頻度を指定しない場合、現行設定値が未変更のまま残ります。索引の更新中、または検索が実行中の場合には、エラー・メッセージが表示されます。このエラー・メッセージは、索引が現在ロックされていて、変更できないことを示します。

以下の例は、索引の更新頻度を変更します。

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

この例では、月曜日から金曜日の 12:00 または 15:00 に、100 個以上のテキスト文書がキューに入れられている場合に、索引が更新されます。

索引の定期的な更新を停止するには、次のコマンドを入力します。

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

あるディレクトリーから別のディレクトリーに索引をコピーする場合、この処理中に索引はロックされます。しかし、コピー後に索引はアンロックされ、再び使用できるようになります。

索引イベントのクリア

サマリー

時期 索引のイベント表内のメッセージが不要になった場合。

コマンド

CLEAR EVENTS FOR INDEX

権限 表に対する CONTROL

更新の開始時刻と終了時刻、索引付き文書の数、または更新時に発生した文書エラーなどの索引作成イベントに関する情報が、索引のイベント表に保管されます。この情報によって、問題の原因を判別できます。これらのメッセージが不要になった場合には、削除できます。

以下の例は、指定したテキスト索引からメッセージを削除します。

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

テキスト索引のドロップ

サマリー

時期 テキスト列でテキスト検索を行う必要がなくなった場合。

コマンド

```
DROP INDEX FOR TEXT
```

権限 表に対する CONTROL

例：

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

テキスト索引をドロップすると、以下の表とビューもドロップされます。

- ログ表とビュー
- イベント表とビュー
- ログ表のトリガー (存在する場合)

注

表をドロップする前に、表の索引を必ずドロップしてください。表を最初にドロップすると、索引が残ってしまいます。

テキスト索引状況の表示

データベース内の現行テキスト索引に関する情報を得るには、ビューを使用します。たとえば、現行データベースのデフォルトを知りたい場合は、以下のコマンドを使用します。

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

現在使用できる索引、それに対応する表、および索引作成の行われた文書に関する情報を得るには、以下のコマンドを使用します。

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

特定の索引のフォーマットに関する情報を得るには、次のようにします。

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

詳細については 199 ページの『付録 C. Net Search Extender インフォメーション・カタログ』を参照してください。

COMMITCOUNT を設定しない場合、db2ext.textindexes からの NUMBER_DOCS パラメータは更新されません。更新処理中に、更新された文書の数を表示するには、以下のコマンドを使用してください。

テキスト索引状況の表示

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

詳細については 98 ページの『CONTROL』を参照してください。

索引のバックアップとリストア

使用可能なデータベースおよび DB2 Net Search Extender によって作成されたテキスト索引をバックアップするには、以下のステップを使用します。

1. DB2 Net Search Extender が作成した索引とその保管場所を探すために、db2ext.textindexes ビュー上で SELECT ステートメントを呼び出します。
db2 "select indschema, indname, indexdirectory from db2ext.textindexes"
2. 索引更新が実行中でないことを確認してから、次のコマンドを使用して DB2 Net Search Extender サービスを停止します。
db2text stop
3. データベースをバックアップした後に、索引ディレクトリーとサブディレクトリーをバックアップします。
4. 次のコマンドを使用して、DB2 Net Search Extender サービスを再始動します。
db2text start

使用可能なデータベースおよび DB2 Net Search Extender によって作成されたテキスト索引をリストアするには、以下のステップを使用します。

1. 次のコマンドを使用して、DB2 Net Search Extender を停止します。
db2text stop
2. 索引ディレクトリーのバックアップ・コピーを、以前と同じパスにリストアします。
3. 次のコマンドを使用して DB2 Net Search Extender を再始動します。
db2text start

/tmp ディレクトリーからのファイルの除去

Net Search Extender サービスを実行中は、以下のファイルのいずれも /tmp ディレクトリーから除去することが許可されていません。

- セマフォア・ファイルおよび共用メモリー・ファイル:

```
CTE* ファイル  
<instance_owner>TEXT.0000.LATCH  
<instance_owner>TEXT.0000  
<instance_owner>CACHE.0000  
<instance_owner>SCHEDULER.LATCH  
<instance_owner>DEMON.SEM  
<instance_owner>DEMON:MEM
```

- 索引の作成中にキャッシュが一時的である場合:

```
<database_name>.IX123456  
<database_name>.IX123456.data0
```

第 7 章 DB2 コントロール・センターの使用

DB2 Net Search Extender 管理関数、DB2 インスタンス、データベース、および、表、ビュー、ユーザー・グループなどのデータベース・オブジェクトを管理するには、DB2 コントロール・センターを使用します。

以下のような、さまざまな DB2 コントロール・センター・オブジェクト上でコマンドを呼び出すことができます。

- インスタンス・オブジェクト (Instance objects)
- データベース・オブジェクト (Database objects)
- 索引オブジェクト (Index objects)

DB2 コントロール・センターの主要なエレメントは、メニュー・バー、ツールバー、オブジェクト・ツリー、および内容ペインです。

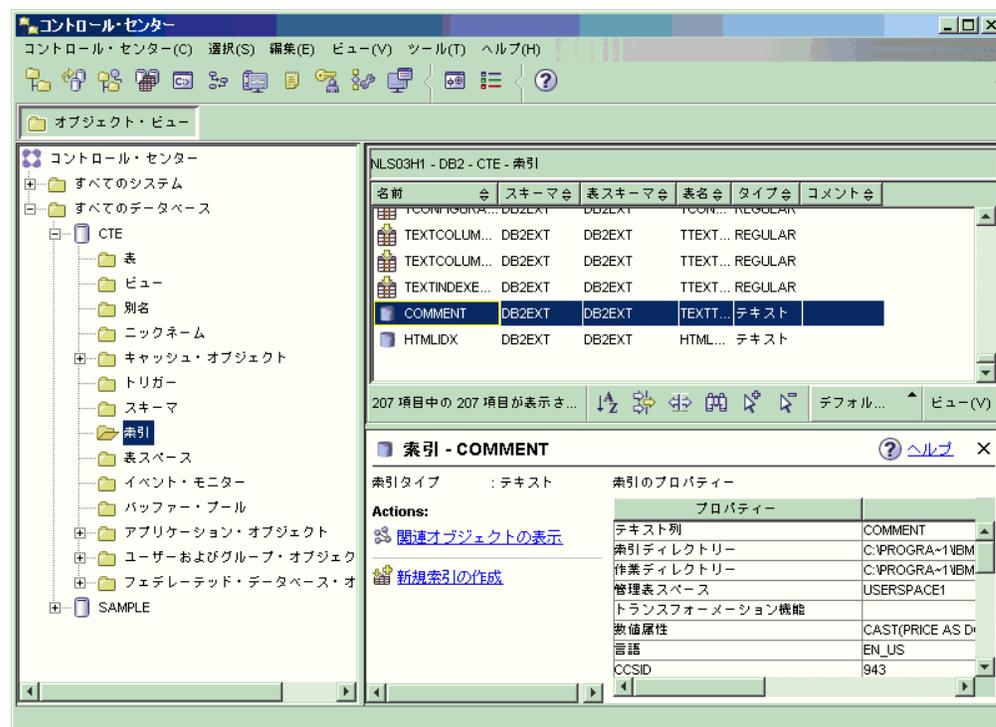


図 6. DB2 コントロール・センター

あるいは、コマンド行を使用できます。詳しくは、以下の章を参照してください。

- 31 ページの『第 6 章 テキスト索引の作成と保守』
- 27 ページの『第 5 章 Net Search Extender インスタンス・サービス』

注

この章の例、および Net Search Extender 用の DB2 コントロール・センターを使用するためには、有効な Net Search Extender ライセンスをデータベース・サーバーにインストールする必要があります。

この章では、DB2 Net Search Extender の索引作成関数および管理関数のみを説明します。DB2 コントロール・センターの使用法については、viii ページの『関連情報』を参照してください。

DB2 Net Search Extender インスタンス・サービスの開始および停止

オブジェクト・ツリーから、使用可能なインスタンスを表示するシステムをクリックします。インスタンスを強調表示し、右クリックして、インスタンス・オブジェクトのポップアップ・メニューを表示します。**Net Search Extender** を強調表示し、ポップアップ・メニューから以下のコマンドのいずれかを選択します。

DB2 Net Search Extender インスタンス・サービスを開始

インスタンス・サービスがまだ開始されていない場合、このコマンドによって開始されます。

DB2 Net Search Extender インスタンス・サービスを停止



図7. 「Net Search Extender サービスの停止 (Stop Net Search Extender Services)」ダイアログ

このコマンドによって、ダイアログが表示されます。インスタンス・サービスおよび索引プロセスを停止するには、チェック・ボックスを使用してください。コマンド構文では、これは FORCE オプションと呼ばれています。

「OK」ボタンをクリックします。

インスタンス状況 (Instance status)

インスタンス状況を示すダイアログを表示します。

詳しくは、97 ページの『第 11 章 インスタンス所有者用の管理コマンド』を参照してください。

データベースを使用可能にする

オブジェクト・ツリーからインスタンス・オブジェクトをクリックして、使用可能なデータベースを表示します。データベースを強調表示し、右クリックしてポップアップ・メニューを表示します。 **Net Search Extender** を強調表示し、拡張メニューから以下のコマンドのいずれかを選択します。

テキストに対しデータベースを使用可能にする

データベースが使用可能になっていない場合にダイアログを表示します。

「OK」 ボタンをクリックして、データベースを使用可能にします。データベースが使用可能になっている場合は、メッセージ・ボックスが表示されません。

テキストに対しデータベースを使用不可にする

データベースが使用不可になっていない場合にダイアログを表示します。

「OK」 ボタンをクリックして、データベースを使用不可にします。データベースがすでに使用不可になっている場合は、メッセージ・ボックスが表示されません。

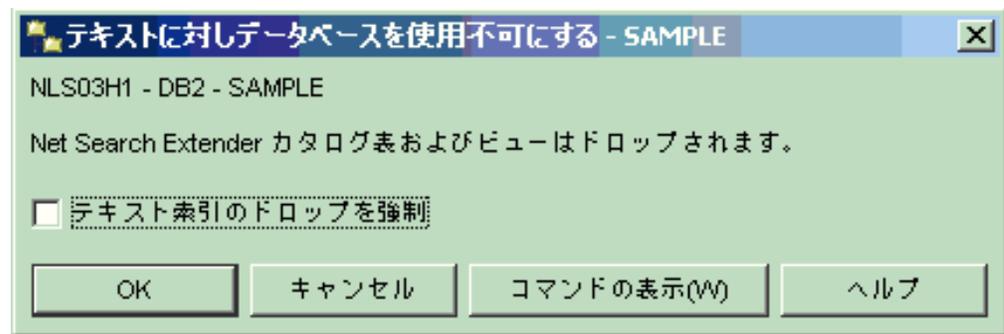


図8. 「テキストに対しデータベースを使用不可にする」ダイアログ

データベースを使用不可にし、すべてのテキスト索引をドロップしたい場合には、チェック・ボックスをクリックします。

データ・リンクの結果サイズの変更

現行のデータ・リンクの結果サイズを示すダイアログを表示します。新規データ・リンク値をキロバイト (KB) 単位で入力し、「OK」 ボタンをクリックします。

詳しくは、103 ページの『第 12 章 データベース管理者用の管理コマンド』を参照してください。

すべてのダイアログにおいて、「コマンドの表示」ボタンは、代替手段としてのコマンド行を表示します。

テキスト索引の管理

オブジェクト・ツリーのデータベース・オブジェクトの下に、索引オブジェクトが表示されます。索引オブジェクトをクリックして、索引を内容ペインに表示します。「Type」列に「text」と入力することにより、テキスト索引オブジェクトを識別できます。

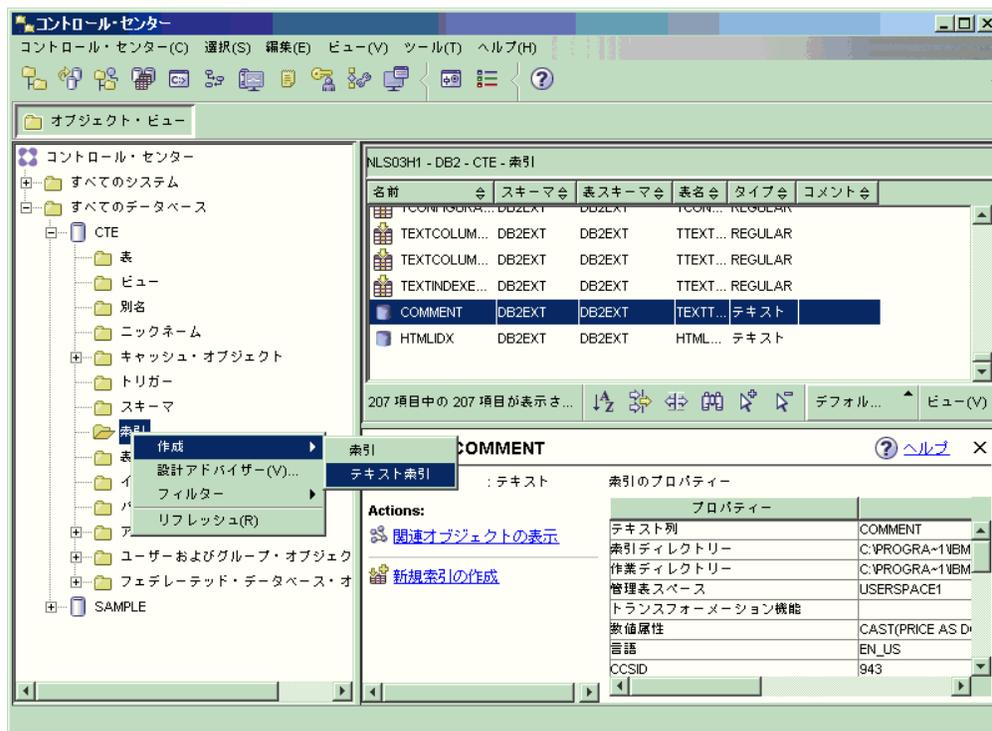


図9. DB2 コントロール・センター

索引オブジェクトで右クリックして、ポップアップ・メニューから以下のコマンドのいずれかを選択します。

作成 このコマンドによって、ダイアログが表示されます。ダイアログで「テキスト索引」を選択することにより、テキスト索引を作成するウィザードを開始します。詳細については、55 ページの『テキスト索引の作成』を参照してください。

フィルター

コントロール・ペイン・ビューに表示する索引オブジェクトを選択するためのダイアログを表示します。

リフレッシュ

オブジェクト・ツリーおよびコントロール・ペイン内の情報を最新表示します。

テキスト索引の保守については、66 ページの『テキスト索引の保守』を参照してください。

注

右クリック・オプションを使用しないで、インスタンス、データベースおよびテキスト索引オブジェクトの各コマンドにアクセスするには、「**選択**」メニュー・コマンドをクリックし、「**Net Search Extender**」を強調表示して、関係のあるコマンドにアクセスします。

テキスト索引を作成する前に、23 ページの『第 4 章 計画』にある前提条件を確認してください。

索引付けに関するその他の前提条件には、以下のものがあります。

- DB2 Net Search Extender インスタンス・サービスの開始
- データベースの使用可能化

テキスト索引の作成

作成コマンドを選択し、拡張メニューで「**テキスト索引**」を強調表示します。「テキスト索引の作成ウィザード」がポップアップ表示されます。このウィザードのいくつかのパネルを使用して、テキスト索引の構成オプションを指定します。

パネル間を移動するには、「**完了**」ボタンが使用可能になるまで、すべての必須情報を入力して、「**次へ**」ボタンをクリックします。テキスト索引を作成するには、「**完了**」ボタンをクリックします。

ビューにテキスト索引を作成するには、120 ページの『CREATE INDEX』で説明されている CREATE INDEX コマンドを使用します。

「名前」パネル

このパネルによって、テキスト索引のスキーマと名前を指定できます。テキスト索引ファイルのための作業ディレクトリーと索引ディレクトリーも指定できます。管理表スペース上に、索引の管理表を作成します。

テキスト索引の作成

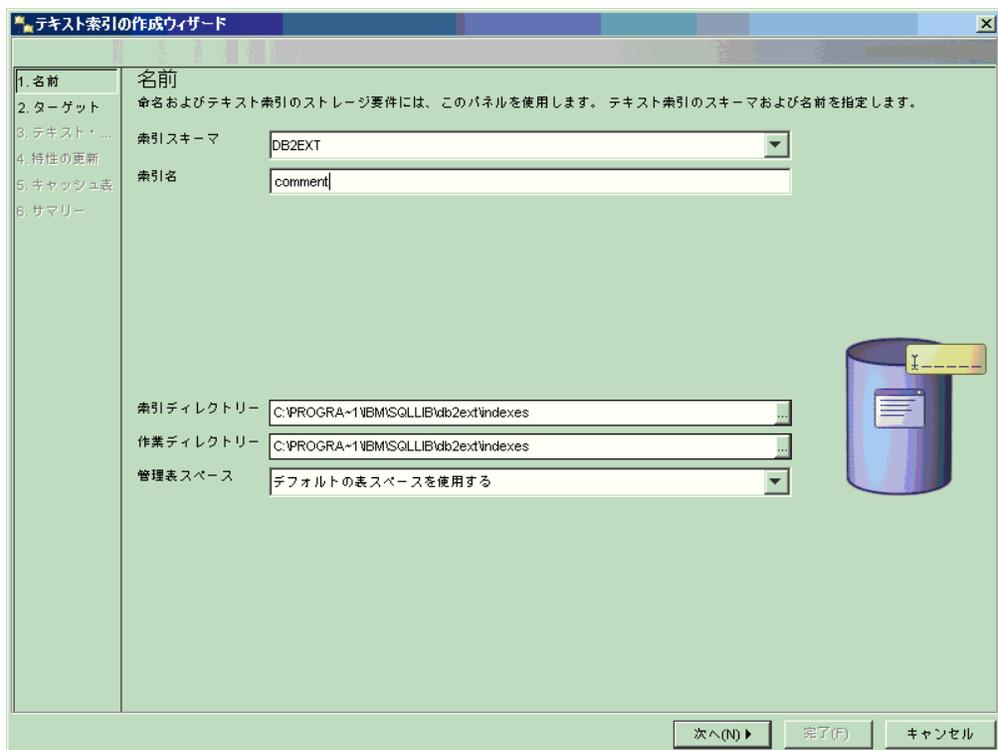


図 10. 「テキスト索引の作成ウィザード」: 「名前」パネル

次は、パネル内の各フィールドについての説明です。

表 1. 「名前」パネルのテキスト・フィールド

フィールド名	必須/ オプション	デフォルト	説明
索引スキーマ	必須	ユーザー ID	テキスト索引のスキーマ名を選択します。これは、索引固有の管理表の DB2 スキーマ名です。
索引名	必須	N/A	テキスト索引用の有効な DB2 索引名を入力します。この名前は、索引スキーマと共にデータベース内のフルテキスト索引を一意に識別します。
索引ディレクトリ	オプション	パス名を参照	テキスト索引の保管先ディレクトリ・パスを指定します。このディレクトリは、DB2 インスタンス所有者のユーザー ID に対する読み取り/書き込み/実行の各権限付きで存在する必要があります。
作業ディレクトリ	オプション	パス名を参照	検索操作と管理操作中に一時ファイルを保管先作業ディレクトリを指定します。このディレクトリは、DB2 インスタンス所有者のユーザー ID に対する読み取り/書き込み/実行の各権限付きで存在する必要があります。
管理表スペース	オプション	デフォルトの表スペースを使用	テキスト索引の管理表の表スペース名を選択します。この表スペースは、ユーザー表の表スペースと同じノード・グループ上に定義する必要があります。

「ターゲット」パネル

このパネルによって、表またはニックネーム表のスキーマと名前、および索引付けしたいデータを含むテキスト列の名前を指定できます。トランスフォーメーション機能を使用して、テキスト列の内容を変更できます。テキスト列に加えて、表の列式の内容に索引付けしたい場合には、数値属性を指定することもできます。

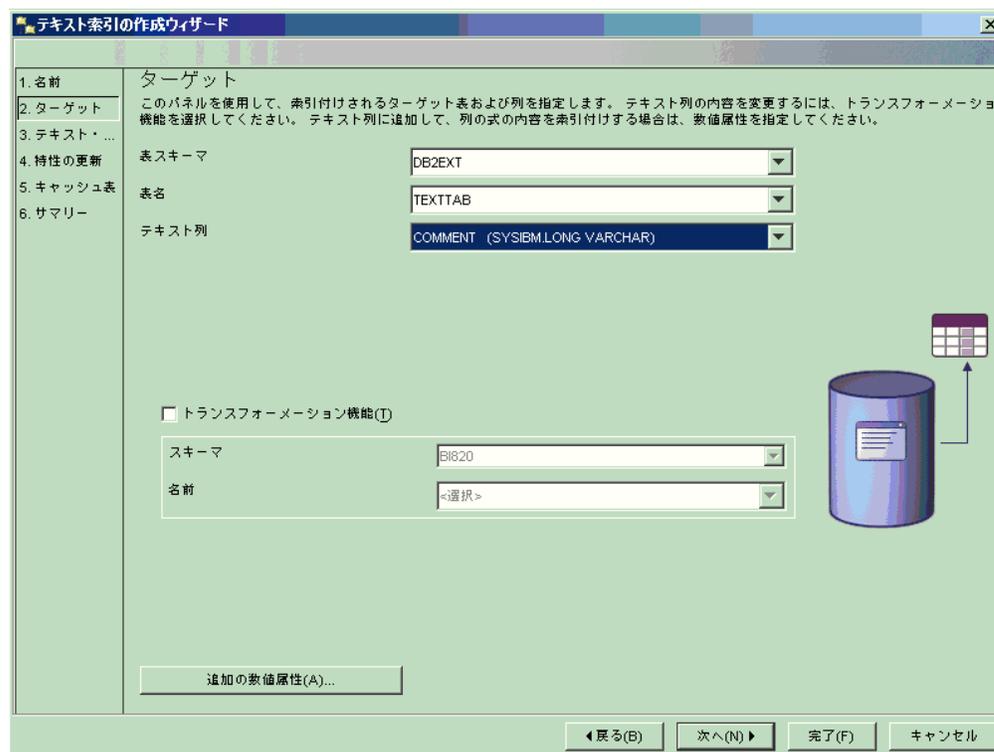


図 11. 「テキスト索引の作成ウィザード」：「ターゲット」パネル

次は、パネル内の各フィールドについての説明です。

表 2. 「ターゲット」パネルのテキスト・フィールド

フィールド名	必須/ オプション	デフォルト	説明
表スキーマ (1)	必須	ユーザー ID	テキスト索引を作成している表またはニックネーム表のスキーマを選択します。
表名 (2)	必須	N/A	索引を作成している表またはニックネーム表の名前を選択します。表には、主キーが必要です。
テキスト列 (3)	必須	N/A	テキスト索引を作成するために使用する列の名前を選択します。列は、以下のいずれかのタイプにトランスフォームするか、またはそのいずれかのタイプである必要があります。すなわち、CHAR (ビット・データの場合)、 VARCHAR (ビット・データの場合)、 LONG VARCHAR (ビット・データの場合)、 CLOB、 DBCLOB、 BLOB、 GRAPHIC、 VARGRAPHIC、 LONG VARGRAPHIC、 および DATALINK。
トランスフォーメーション機能	オプション	使用不可	変換関数を使用するために選択します。

テキスト索引の作成

表 2. 「ターゲット」 パネルのテキスト・フィールド (続き)

トランスフォーメーション機能: スキーマ	必須 (機能が選択されている場合)	ユーザー ID	テキスト文書にアクセスするために使用する UDF のスキーマを選択します。
トランスフォーメーション機能: 名前	同上	N/A	テキスト文書にアクセスするために使用する UDF の名前を選択します。

「表スキーマ (1)」、 「表名 (2)」、 および「テキスト列 (3)」は、この順序でのみ指定できます。



図 12. 「数値属性」 ダイアログ

属性を表示または追加するには、「追加の数値属性」ボタンをクリックします。ウィンドウが表示されます。数値属性を索引に追加するには、「追加」ボタンをクリックします。すると、さらにウィンドウが表示されます。SQL 列式と属性の名前を指定します。

別の方法では、属性を選択し、該当するボタンを押して、項目を変更、移動または除去します。

数値属性の説明

テキスト列に加えて、列式に索引付けたい場合には、数値属性を指定します。例：テキスト列に加えて、タイプ `TIMESTAMP` の列日付に索引付けたい場合は、数値属性 `"cast(julian_day(date) as double)"` を指定し、属性の名前を指定します。

検索照会内で数値表現を使用したい場合は、数値属性を指定します。SQL 照会を使用して検索する場合は、数値属性を使用する代わりに結合検索を使用できます。例：`WHERE numattrib = 123 AND contains('...')` のようにします。

「テキスト・プロパティ」パネル

このパネルによって、テキスト文書の言語とフォーマットを指定できます。文書がデータベースと同じ CCSID でなく、テキスト列がバイナリー形式の場合には、その CCSID を指定します。データベースの CCSID は、最初に選択していることに注意してください。文書が GPP、HTML、Outside-In、または XML 構造化フォーマットであれば、文書モデルを指定できます。

注

フォーマット・リスト・ボックスでは、Outside-In フィルター・フォーマットは INSO と記載されています。

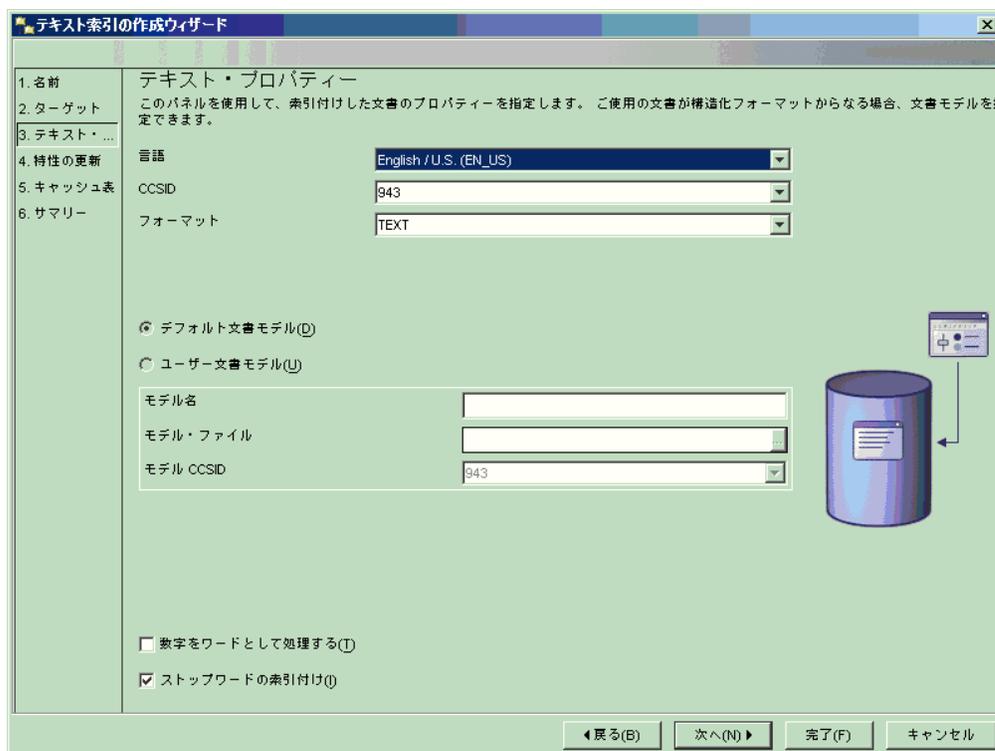


図 13. 「テキスト索引の作成ウィザード」: 「テキスト・プロパティ」パネル

次は、パネル内の各フィールドについての説明です。

表 3. 「テキスト・プロパティ」パネルのテキスト・フィールド

フィールド名	必須/ オプション	デフォルト	説明
言語	オプション	EN_US	言語を選択することによって、文書に索引を付ける場合に、文の終わりと段落の終わりの区切り文字を判別できるようにします。
CCSID	オプション	データベースの CCSID	テキスト文書に索引を付けるための CCSID を選択します。
フォーマット	オプション	TEXT	HTML、XML、TEXT、INSO、または GPP からテキスト文書フォーマットを選択します。

テキスト索引の作成

表 3. 「テキスト・プロパティ」 パネルのテキスト・フィールド (続き)

デフォルトの文書モデル	オプション	使用可能	デフォルトの文書モデルを使用します。
ユーザー文書モデル	オプション	使用不可	ユーザーの文書モデルを使用します。
モデル名	必須 (ユーザー文書モデルが選択されている場合)	N/A	文書モデルの名前を入力します。 HTML、XML、Outside-In、および GPP フォーマットの場合は、文書モデルを指定できます。名前は、モデル・ファイル内でのみ検出されることに注意してください。
モデル・ファイル	同上	N/A	文書モデル・ファイルを指定します。ファイルは DB2 インスタンス所有者にとって、読み取り可能でなければなりません。
モデル CCSID	同上	データベースの CCSID	CCSID を選択して、文書モデル・ファイルの内容を解釈できるようにします。
数字をワードとして処理する	オプション	使用不可	数字のシーケンスを、文字に隣接している場合でも、別個のワードとして解釈するために選択します。
索引ストップワード	オプション	使用可能	言語特有のストップワード処理を使用可能にするために選択します。ディレクトリー <code>sql1lib/db2ext/resources</code> 内の <code><language>.tsw</code> に、ストップワード・リストが含まれています。

「特性の更新」 パネル

このパネルによって、索引を増大して更新するか、または最初から再作成するかを指定できます。指定した時刻に索引が自動的に更新されるように更新設定を指定できます。



図 14. 「テキスト索引の作成ウィザード」: 「特性の更新」パネル

次は、パネル内の各フィールドについての説明です。

表 4. 「特性の更新」パネルのテキスト・フィールド

フィールド名	必須/ オプション	デフォルト	説明
インクリメンタル更新	オプション	使用可能	インクリメンタル索引更新用に選択します。チェック・ボックスを使用可能にしない場合は、更新操作が実行されるときに索引を再作成してください。
コミット・カウント	オプション	0	1 つのトランザクションでの更新中に処理される変更の数。 コミット・カウントはパフォーマンスに影響します。詳しくは、45 ページの『パフォーマンスに関する考慮事項』を参照してください。
表特性のキャプチャー	オプション	N/A	ソース表の変更をキャプチャーするために、レプリケーション・キャプチャー表を使用する場合に選択します。レプリケーション・キャプチャー表は、変更データ (CD) 表か整合変更データ (CCD) 表のいずれかでなければならず、DB2 Net Search Extender 生成のログ表を置き換えます。
スキーマ名のレプリケーション・キャプチャー	オプション	ユーザー ID	レプリケーション・キャプチャー表のスキーマ名。表は、DB2 レプリケーションを使用して事前に作成しておく必要があります。

テキスト索引の作成

表 4. 「特性の更新」パネルのテキスト・フィールド (続き)

表名のレプリケーション・キャプチャー	「表特性のキャプチャー」が使用可能になっている場合は必須	N/A	レプリケーション・キャプチャー表の表名表は、DB2 レプリケーションを使用して事前に作成しておく必要があります。
表のスキーマ名の制御	「表特性のキャプチャー」が使用可能になっている場合は必須	N/A	表のスキーマ名の制御。表は、DB2 レプリケーションを使用して事前に作成しておく必要があります。
REORG (「自動」または「手動」ラジオ・ボタン)	オプションまたは必須	使用可能/使用不可	索引再編成を自動または手動で実行します。
更新のための変更の最小数	オプション	1	指定された時刻に索引をインクリメンタル更新するには、テキスト文書に少なくともいくつかの変更が加えられていることが必要かを指定します。
更新スケジュール	オプション	使用不可	自動更新設定値を追加するために選択します。

索引更新の設定値を追加するには、「設定」ボタンをクリックします。このボタンは、「更新スケジュール」を選択した場合のみ使用可能になります。ダイアログで、更新時刻の日、時、分を選択します。複数の日を選択した場合、更新は選択したすべての日の同じ時刻に行われます。

「キャッシュ表」パネル

このパネルによって、索引に加えてキャッシュ表を指定できます。キャッシュする結果列を指定することができます。キャッシュは、ストアード・プロシージャを使用して検索できます。タイプ、最大サイズ、および初期索引作成時にユーザー表の内容を検索する順序などの、他のキャッシュ・パラメーターも指定できます。

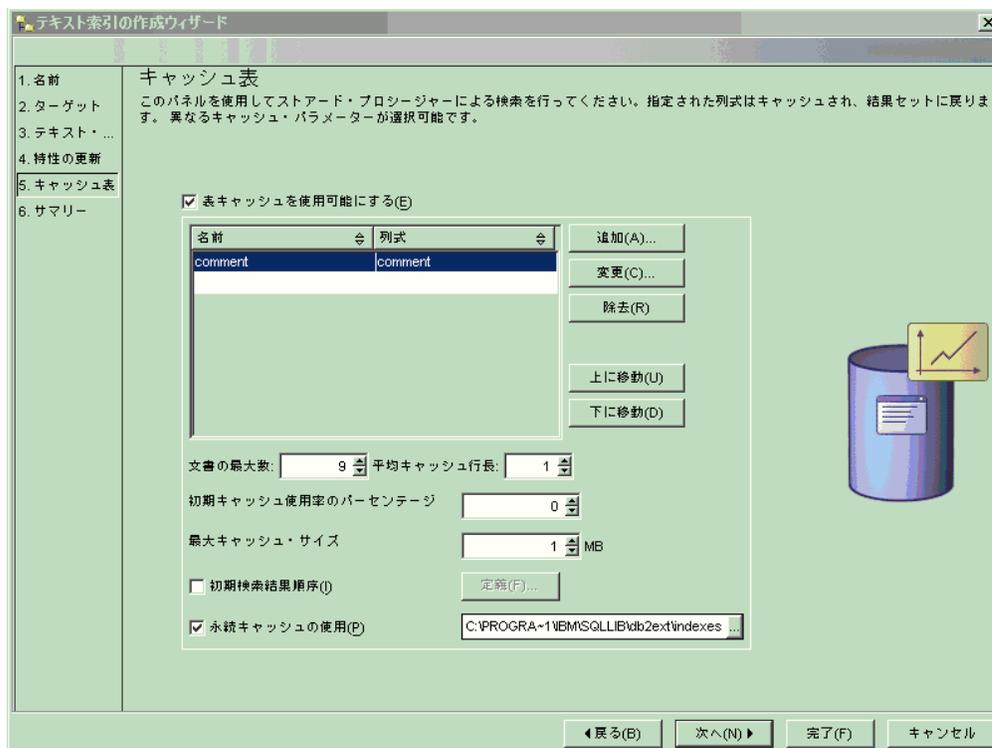


図 15. 「テキスト索引の作成ウィザード」: 「キャッシュ表」パネル

次は、パネル内の各フィールドについての説明です。

表 5. 「結果キャッシュ (Result Cache)」パネルのテキスト・フィールド

フィールド名	必須/ オプション	デフォルト	説明
表キャッシュを使用可能にする	オプション	使用不可	キャッシュ表の作成を使用可能にするために選択します。
結果列表 (Result column table)	必須 (「表キャッシュを使用可能にする」が選択されている場合)	N/A	検索結果列を指定する SQL 列式のリストが表示されます。
文書の最大数	必須	表の行数	『キャッシュ使用率およびキャッシュ・サイズの決定』のセクションを参照してください。
平均キャッシュ行長	必須	N/A	『キャッシュ使用率およびキャッシュ・サイズの決定』のセクションを参照してください。
初期キャッシュ使用率のパーセンテージ	オプション	50%	追加の文書のためにフリーにしておくキャッシュのパーセンテージを選択します。
最大キャッシュ・サイズ	オプション	N/A	索引をアクティブにするときに作成するキャッシュ表の最大サイズを指定します。この数値が小さすぎると、アクティブ化は失敗します。
初期検索結果順序	オプション	使用不可	検索結果の順序の定義を選択します。文書はキャッシュ表内と同じ索引付け順序で戻されます。この順序は、インクリメンタル更新後には保証されません。

表 5. 「結果キャッシュ (Result Cache)」 パネルのテキスト・フィールド (続き)

永続キャッシュの使用	オプション	使用可能	このオプションは、非アクティブ化またはシステム・リブート後の高速アクティブ化の実行を可能にします。永続キャッシュにはディレクトリー・パスを指定する必要があることに注意してください。キャッシュを一時的なものにする必要がある場合は、使用不可にしておいてください。
------------	-------	------	---

— キャッシュ使用率およびキャッシュ・サイズの決定 —

「初期キャッシュ使用率のパーセンテージ」は、追加の文書のためにフリーにしておくキャッシュのパーセンテージを指定します。「最大キャッシュ・サイズ」は、キャッシュのアクティブ化の際に作成するキャッシュ表の最大サイズを指定します。これらのオプションは、以下の要因によって異なります。

- 表にある実際の文書数。
- 予想更新数。
- キャッシュに入れたい SQL 式の平均サイズ。

「初期キャッシュ使用率のパーセンテージ」および「最大キャッシュ・サイズ」に推奨値を入力できます。あるいは、「文書の最大数」または「平均キャッシュ行長」に値を入力するたびに、再計算させることもできます。

「文書の最大数」の値は、最初は、表の行数に設定されます。この値は、文書の数および予想変更数に従って変更してください。文書のすべての更新数、追加数、および削除数を含めます。

キャッシュ表リストに SQL 式を追加する場合は、結果の長さに従って、「平均キャッシュ行長」が計算されます。これは表の中の行数を基にしているため、計算に相当の時間を要することがあります。この平均値が小さい場合は、値を変更してください。

たとえば、表に 10 個の項目があり、列式の合計が 100 である場合は、これらの値を最初に設定します。文書の最大数 (削除された文書を含む) が 10 000 であると予測した場合は、その数値を入力します。平均の列式の数が、VARCHAR(100) などの計算値より小さく、テキスト・サイズが 10 と入力されている場合は、平均行サイズとしてこの計算値を使用します。

初期検索結果順序を定義するには、「定義」ボタンをクリックします。このボタンは、「初期検索結果順序」チェック・ボックスを選択した場合のみ使用可能であることに注意してください。ダイアログに、すべての指定した SQL 列式が表示されます。結果順序を追加するには、「追加」ボタンをクリックし、ダイアログで SQL の結果順序を指定します。

項目を変更、移動、または除去するには、式を選択し、該当するボタンをクリックします。



図 16. 「初期検索結果順序」ダイアログ

SQL 列式を追加するには、「結果列表 (Result Column table)」の横の「追加」ボタンをクリックします。ダイアログで、結果列の式と名前を指定します。

項目を変更または除去するには、列式をクリックします。すると、該当するボタンが使用可能になります。



図 17. 「列式の変更」ダイアログ

「サマリー」パネル

このパネルは、以前に選択したパラメーターの概要を示します。



図 18. 「テキスト索引の作成ウィザード」: 「サマリー」パネル

「コマンドの表示」ボタンをクリックすると、「完了」ボタンをクリックしたときに実行されるコマンドが表示されます。このアクションによって、テキスト索引が作成されます。

テキスト索引の保守

テキスト索引を保守するには、コントロール・パネルでテキスト索引を選択し、「選択」メニュー・コマンドを選択します。メニューから、以下のコマンドのいずれかを選択できます。

1. ALTER コマンド。テキスト索引を変更するためのコマンド。
2. DROP コマンド。テキスト索引をドロップするためのコマンド。
3. UPDATE コマンド。テキスト索引を更新するためのコマンド。
4. SHOW INDEX EVENTS コマンド。索引イベントを表示するためのコマンド。
5. ACTIVATE INDEX MEMORY コマンド。索引キャッシュをアクティブ化するためのコマンド。
6. DEACTIVATE INDEX MEMORY コマンド。索引キャッシュを非アクティブ化するためのコマンド。
7. SHOW STATUS コマンド。索引状況を表示するためのコマンド。

アクティブ化および非アクティブ化するためのコマンドは、キャッシュ・オプションを指定して索引を作成する場合にのみ表示されます。

テキスト索引の変更

「変更 (Alter)」コマンドを選択すると、ダイアログに一連のパネルが表示されます。これらのパネルは、テキスト索引のパラメーターの概要を示します。必ずしもすべてのパラメーターを変更できるわけではありません。



図 19. 「索引の変更」ダイアログ: 「名前」タブ

「名前」パネルは、索引の名前とストレージ構成を表示します。索引ディレクトリと作業ディレクトリを変更できます。

「ターゲット」パネルは、索引のターゲットおよび数値属性の設定値を表示します。これらの構成は変更できません。

「テキスト」パネルは、テキスト文書の構成を表示します。これらの設定は変更できません。



図 20. 「索引の変更」ダイアログ: 「更新」タブ

「更新」パネルは、索引構成の更新特性を表示します。更新スケジュールを変更できます。索引をインクリメンタル更新用に作成した場合には、変更の最小数も変更できます。索引をコミット・カウント・オプションを使用して作成した場合には、コミット・カウント値も変更できます。



図 21. 「索引の変更」ダイアログ: 「キャッシュ表」タブ

「キャッシュ表」パネルは、キャッシュ・オプションの設定値を表示します。結果キャッシュがすでに使用可能であれば、永続ディレクトリを変更したり、索引キャッシュを一時的に変更することができます。さらに、「最大キャッシュ・サイズ」および「初期キャッシュ使用率のパーセンテージ」を変更することもできます。

テキスト索引のドロップ

「ドロップ」コマンドを選択すると、ダイアログに選択可能なテキスト索引が表示されます。

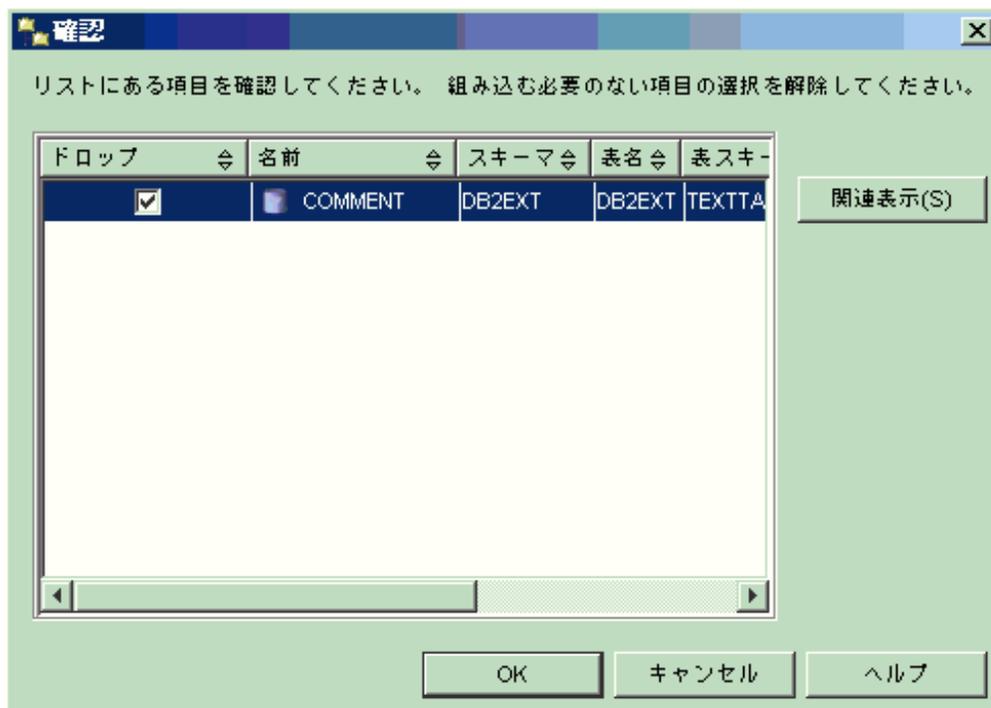


図 22. 「索引のドロップ (Drop Index)」 ダイアログ

索引を選択し、「OK」ボタンをクリックします。

テキスト索引の更新

「更新」コマンドを選択すると、ダイアログに複数の更新オプションが表示されます。



図 23. 「索引の更新」 ダイアログ

更新操作のコミット・カウントを指定できます。索引の作成時または変更時に指定した最小数を更新したい場合には、「変更の最小数」チェック・ボックスを選択してください。索引を再編成するには、このチェック・ボックスを選択します。再編成が推奨される場合には、このチェック・ボックスが使用可能にされることに注意してください。

索引イベントの表示

「索引イベントの表示 (Show index events)」コマンドを使用すると、イベント表の内容がダイアログに表示されます。



図 24. 「索引イベント」ダイアログ

索引イベントをクリアするには、「リセット」ボタンをクリックしてください。

テキスト索引キャッシュのアクティブ化

「索引メモリのアクティブ化 (Activate index memory)」コマンドを選択すると、ダイアログが表示されます。



図 25. 「キャッシュのアクティブ化」ダイアログ

キャッシュをアクティブ化するには、「OK」ボタンをクリックします。キャッシュを最初から作成した場合は、チェック・ボックスを選択してください。

テキスト索引キャッシュの非アクティブ化

「索引メモリの非アクティブ化 (Deactivate index memory)」コマンドを選択すると、ダイアログが表示されます。

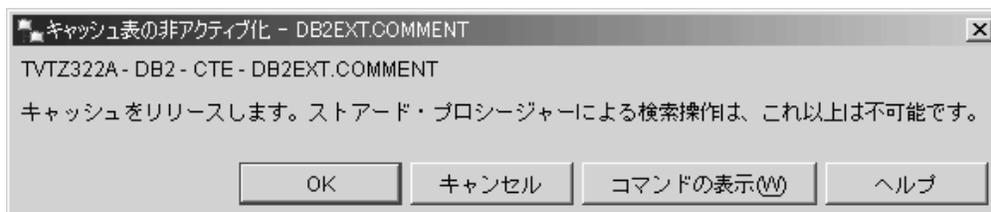


図 26. 「キャッシュ表の非アクティブ化」ダイアログ

キャッシュを解放するには、「OK」ボタンをクリックします。

索引状況の表示

「状況表示 (Show status)」コマンドを選択すると、ダイアログが表示されます。

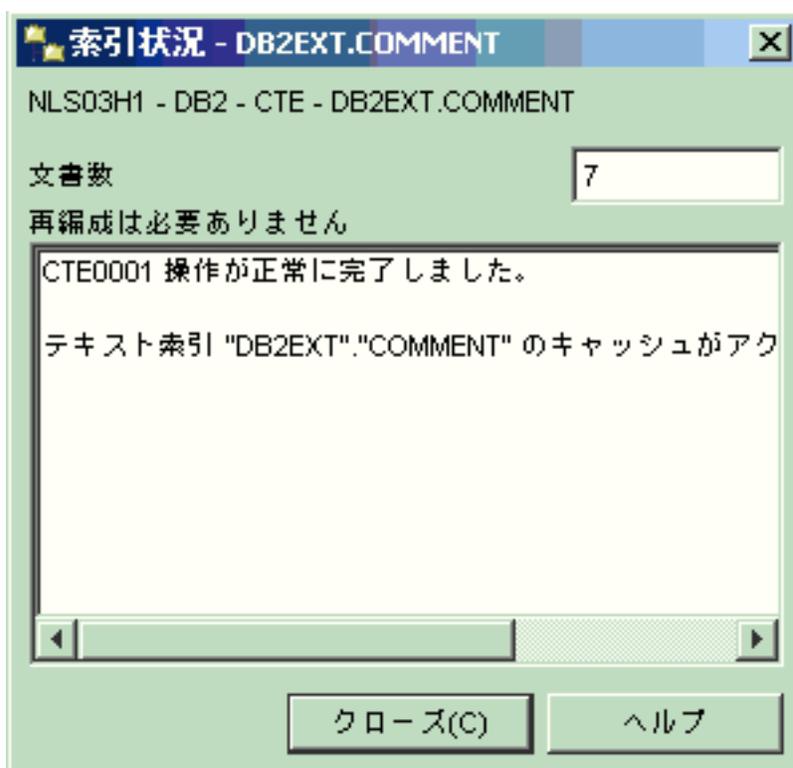


図 27. 「索引状況」ダイアログ

これには、索引付けされた文書の数、再編成推奨フラグおよびその他の索引情報が含まれます。

第 8 章 検索

DB2 Net Search Extender には、テキストを検索するための以下の方式があります。

SQL スカラー検索関数

この関数を使用すると、テキスト検索副照会を SQL 照会に組み込めるようになります。Net Search Extender は、SQL で通常使用可能な関数に加えてこれらの関数を提供しています。

ストアード・プロシージャ検索関数

これを使用すると、定義済みのキャッシュ結果表を戻すことができます。

SQL 表値関数

この検索は、ストアード・プロシージャでの検索と同じように使用できます。

SQL スカラー検索関数について、この章では以下の項目を説明します。

- CONTAINS、NUMBEROFMATCHES、および SCORE 関数を使用する、テキストの検索。

構文の説明は 155 ページの『第 15 章 SQL スカラー検索関数および SQL 表値関数』を参照してください。

- CONTAINS 関数の例を使用した検索引き数の指定。

構文の説明は 147 ページの『第 14 章 検索引き数の構文』を参照してください。

ストアード・プロシージャ検索関数について、この章では以下の項目を説明します。

- ストアード・プロシージャ検索を使用してテキストを検索する。
- 検索引き数の指定については、147 ページの『第 14 章 検索引き数の構文』のパラメーターの説明を参照してください。

SQL 表値関数について、この章では以下の項目を説明します。

- SQL 表値関数および HIGHLIGHT 関数を使用するテキストの検索。

構文の説明は 155 ページの『第 15 章 SQL スカラー検索関数および SQL 表値関数』を参照してください。

- 検索引き数の指定については、147 ページの『第 14 章 検索引き数の構文』のパラメーターの説明を参照してください。

この章では、考慮する必要があるかもしれない検索パフォーマンスについても説明します。

検索する前に、異なるデータ・タイプを含め、31 ページの『第 6 章 テキスト索引の作成と保守』で説明したすべての該当する索引作成のステップを考慮に入れるようにしてください。

注

システム・シェルは、?、(、)、*、!、および " などの特殊文字を解釈します。したがって、コマンドにこれらの文字が含まれている場合には、引用符またはエスケープ文字を使用する必要があります。

以下に、特殊文字を使用する UNIX コマンドの例を示します。

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

SQL スカラー検索関数を使用するテキストの検索

このセクションでは、以下の方法で SQL スカラー検索関数を使用する方法を、例を挙げて説明します。

- 関数 CONTAINS を使用して照会を行う。
- 関数 NUMBEROFMATCHES を使用してテキスト文書内の一致の数を判断する。
- 関数 SCORE を使用して検出されたテキスト文書の適合性を入手する。

構文の説明は 155 ページの『第 15 章 SQL スカラー検索関数および SQL 表値関数』を参照してください。

照会を行う

この例は、CONTAINS 関数が、表 texttab の列 comment 内のテキストを検索する方法を示しています。この関数は、テキストが検索引き数を満足する場合には 1 を返し、それ以外の場合には 0 を返します。

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

この例では、列 COMMENT で用語 book を検索します。

注

パフォーマンスを向上させるには、制限的な検索基準を追加するのが有益です。例を挙げます。

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

検索を実行し、検出された一致の数を返す

NUMBEROFMATCHES 関数を使用して、各テキスト文書で検索値が検出される頻度を判断します。

```
SELECT AUTHOR,TITLE,
      NUMBEROFMATCHES(COMMENT, "book")
      FROM DB2EXT.TEXTTAB
```

NUMBEROFMATCHES によって整数値が戻されます。

検索を実行し、検出されたテキスト文書のスコアを戻す

SCORE は、この文書が、検出された他の文書と比較して、検索値にどの程度一致しているかを示す絶対値です。この値は、文書内で検出された一致の数を、文書のサイズとの関係で示します。以下の例では、SCORE 関数を使用して、検出された文書のスコアを入手しています。

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

SCORE は、0 と 1 の間の DOUBLE 値を戻します。

注

ビューに作成される索引については、CONTAINS、SCORE、および NUMBEROFMATCHES 検索関数は使用できません。

分散 DB2 環境では、SCORE 値は異なります。

- 非分散環境では、すべての文書が 1 つの表の中にあります。SCORE 値は、単一の表、および表内の他のすべての文書との文書関係に基づいたものになります。
- 分散 DB2 環境では、すべての文書が異なるノードにあります。索引の作成中には、ローカル文書だけがテキスト索引の作成に使用されます。これは、各ノード上のローカルな文書です。この場合、SCORE 値は、複数ノードのうちただ 1 つのノードにある全文書との文書関係に基づいたものになります。

SQL 検索引き数の指定

CONTAINS、NUMBEROFMATCHES、および SCORE の各関数はすべて、検索引き数を使用します。このセクションでは、CONTAINS 関数を使用して、DB2 Net Search Extender 関数での検索引き数のさまざまな例を示します。

構文の説明は 147 ページの『第 14 章 検索引き数の構文』を参照してください。

任意の順序での用語検索

1 つの検索引き数に複数の用語を指定できます。複数の検索語を結合する 1 つの方法では、次のようにコンマを使用して結合します。

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

この形式の検索引き数では、いずれかの検索語を含んでいるテキストが検索されます。論理学の用語で言えば、OR 演算子で検索語を接続します。

ブール演算子 AND と OR を使用する検索

検索語は、ブール演算子 “&” (AND) および “|” (OR) を使用して、他の検索語と結合できます。

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

ブール演算子を使用して複数の検索語を結合することもできます。

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

複数のブール演算子を使用する場合は、左から右に評価されます。ただし、論理 AND 演算子 (&) は、論理 OR 演算子 (|) より強く結合します。以下の例ではこの評価が示されています。この例では、括弧は含まれていません。

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

括弧がないので、Net Search Extender は、ブール演算子を次のように評価します。

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

ブール演算子を正しく評価するには、次のように括弧を使用する必要があります。

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

次のように、コンマ区切り記号を使用してチェーニングした検索語でブール演算子を結合することもできます。

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

ただしこの場合、コンマは次のようにブール OR 演算子として解釈されます。

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

追加情報については、78 ページの『ブール演算子 NOT を使用する検索』も参照してください。

ファジー検索

「ファジー」検索は、検索語と類似したスペルのワードを検索します。

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

この例では、検索によって、つづりを誤ったワード pulitzer の出現を検出しています。

突き合わせレベル (例では 「80」) は、正確度を指定します。文書内につづりの誤りがある可能性が高い場合は、ファジー検索を使用してください。光学式文字認識装置 (OCR) または音声入力で作成した文書の場合には、つづりの誤りがよく起こります。レベルとして 1 から 100 を使用します。100 は正確な一致を表し、80 以下は「ファジー」を表します。

注

ファジー検索では妥当な正確度が得られない場合は、文字のマスキングを使用して、用語の一部を検索してください。

用語の一部の検索 (文字のマスキング)

「ワイルドカード」文字とも呼ばれるマスク文字は、検索をより柔軟にする方法を提供します。これにより、検索によって検出されるテキスト文書の数が増えます。

DB2 Net Search Extender は、パーセント (%) と下線 (_) という 2 つのマスク文字を使用します。

- % は、**不定数の任意の文字**を表します。検索語の真ん中のマスク文字として % を使用した例を次に示します。

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

この検索語は、“thriller” というワードを含むテキスト文書を検索します。

- _ は、検索語内の **1 文字**を表します。次の例は、“thriller” というワードを含んでいるテキスト文書を検出します。

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

注

ワイルドカード文字は、結果リストのサイズを大幅に増加して、パフォーマンスを低下させ、予期しない検索結果を戻す可能性があるため、控えめに使用してください。

マスク文字をすでに含んでいる用語の検索

「%」文字または「_」文字を含んでいる用語を検索する場合、これらの文字の前にエスケープ文字と呼ばれる文字を置いてください。すると、ESCAPE キーワードを使用してエスケープ文字を識別できます。

次の例では、“!” がエスケープ文字です。

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
      WHERE CONTAINS(COMMENT,
                    '"!100!%" ESCAPE "!"') = 1
```

固定順序での用語検索

“primary key” を検索する場合、2 つの用語が隣接し、指定したシーケンスで存在する場合にのみ、これらの用語が検出されます。

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

同じ文または同じ段落内での用語検索

複数の検索語が同じ文内で存在するテキスト文書を検出する検索引き数の例を、次に示します。

```
SELECT AUTHOR, TITLE
       FROM DB2EXT.TEXTTAB
       WHERE CONTAINS(COMMENT,
                      '"web" IN SAME SENTENCE AS "disk"') = 1
```

一緒に存在する 3 つ以上のワードも検索できます。次の例では、同じ段落で存在する 2 つのワードについての検索が行われます。

```
SELECT AUTHOR, TITLE
       FROM DB2EXT.TEXTTAB
       WHERE CONTAINS(COMMENT,
                      '"computer" IN SAME PARAGRAPH AS "web"') = 1
```

構造化文書のセクション内での用語検索

構造化文書のサブセクション H2 内で、検索語 IBM が現れるテキスト文書を検出する検索引き数の例を、次に示します。

```
SELECT CATEGORY, DATE
       FROM DB2EXT.HTMLTAB
       WHERE CONTAINS(HTMLFILE,
                      'SECTIONS ("H2") "IBM"') = 1
```

セクション名には大文字小文字の区別があることに注意してください。モデル・ファイルと照会でのセクション名が同じになるようにしてください。詳細については、87 ページの『第 9 章 構造化文書を使用する作業』を参照してください。

ブール演算子 NOT を使用する検索

ブール演算子 NOT を使用して、特定のテキスト文書を検索から除外できます。

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

この例では、用語 “book” を含むすべてのテキスト文書を、“author” または “pulitzer” の検索から除外しています。

シソーラス検索

シソーラス検索は、DB2 Net Search Extender での強力な検索語拡張関数です。検索対象となる追加の用語が、ユーザー自身が作成したシソーラスから取られるので、これらの用語を直接制御できます。たとえば、「データベース」を検索するとき、「リポジトリ」と「DB2」のような用語も検出できます。

頻繁に検索し、より効果的な検索結果を生成したい特定の関連エリアでこのタイプの検索を使用します。

シソーラスの作成方法の詳細については、89 ページの『第 10 章 検索項目を拡張するためのシソーラスの使用』を参照してください。次の例は、シソーラス拡張関数を使用するための構文を示しています。

この例は、「product」という用語を指定し拡張することによって、シソーラス“nseamplethes”で検出される、この用語のすべての関連用語を追加します。ここでは、“marketing”が追加され、検索されます。

```
SELECT CATEGORY, DATE
      FROM DB2EXT.HTMLTAB
      WHERE CONTAINS(HTMLFILE,
                    'THESAURUS "nseamplethes"
                    EXPAND RELATED
                    TERM OF "product"') = 1
```

次の例は、「product」という検索語を指定します。すると、検索語のすべての同義語に拡張して検索が実行されます。

```
SELECT CATEGORY, DATE
      FROM DB2EXT.HTMLTAB
      WHERE CONTAINS(HTMLFILE,
                    'THESAURUS "nseamplethes"
                    EXPAND SYNONYM
                    TERM OF "product"') = 1
```

数値属性検索

構造化文書に保管されている数値属性は、次の構文を使用して検索できます。

```
SELECT AUTHOR, TITLE
      FROM DB2EXT.TEXTTAB
      WHERE CONTAINS(COMMENT,
                    'ATTRIBUTE "PRICE" between 9 and 20') = 1
```

属性名には大文字小文字の区別があることに注意してください。モデル・ファイルと照会での属性名が同じになるようにしてください。詳細については、87ページの『第9章 構造化文書を使用する作業』を参照してください。

フリー・テキスト検索

「フリー・テキスト検索」は、検索語がフリー・フォーム・テキストとして表される検索です。検索対象のサブジェクトを、自然言語の句または文によって示します。フリー・テキスト照会でのワードの順序は問題とはなりません。ただし、照会用語のセットのうちの少なくとも1つの用語が、検索対象の文書に出現している必要があります。

フリー・テキスト引き数の検索ストリングでは、文字またはワードのマスキングがサポートされていないことに注意してください。

例：

```
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
```

照会を SCORE 検索関数と結合することにより、検索によってワード「something」だけを含んだ文書も戻されることとなります。

その他の検索構文の例

追加の検索構文の例に慣れるために、`search` という名前のコマンド行プロセッサ入力ファイルを使用します。このファイルには、サンプル表に対して実行する DB2 Net Search Extender の検索関数の例が入っています。

例を実行するには、次の構文を使用します。

```
db2 -tvf search
```

表と索引が作成されていない場合は、次のいずれかを実行します。

- UNIX プラットフォームの場合:
<instance_owner_home>/<sqllib>/samples/db2ext ディレクトリーで `nseample` を実行します。
- Windows プラットフォームの場合: <sqllib>/samples/db2ext ディレクトリーで `nseample (.bat)` を実行します。

ストアード・プロシージャ検索を使用するテキスト検索

アプリケーションがテキスト検索結果のサブセットを、ハイパフォーマンスで必要とする場合には、ストアード・プロシージャ検索インターフェースを使用します。すべての結果が必要な場合、または大量の文書の索引を作成する必要がある場合には、ストアード・プロシージャを使用しないでください。主な理由は、ユーザー表の一部がメモリーにコピーされるため、大量の実メモリーが使用可能でなければならないということなのです。

ストアード・プロシージャでは、まず 0 から 20 の結果を要求し、次に 21 から 40 という具合に、カーソル・ナビゲーションと類似の方法をとることができます。このカーソル機能とキャッシュ (索引作成中に計算される) の使用を組み合わせると、検索は極めて高速となります。これは特に、結合が不要になるためです。

ストアード・プロシージャを使用する場合は、以下のオプションを確認してください。

- 分散 DB2 環境では、ストアード・プロシージャのために単一ノード上の表スペースを明示的に指定し、明示的にこのノードに接続しなければならない。
- CREATE INDEX の実行時に、`Cache-search-result` オプションが指定されている。
- インクリメンタル更新実施の可能性を含めて、現在および将来の共有メモリー所要量が、完全に考慮に入れられている。詳しくは、40 ページの『ストアード・プロシージャ検索で使用できるテキスト索引の作成』を参照してください。
- `db2text activate` コマンドを使用して、索引のキャッシュがアクティブ化されている。

以下は、ストアード・プロシージャ検索の例です。

```
db2 "call db2ext.textSearch('¥"book¥"', 'DB2EXT', 'COMMENT', 0, 2, 1, 1, '?', '?")"
```

最初のパラメーターは、検索語です。構文は、SQL 関数の場合と全く同じです。それから索引名と索引スキーマを指定します。名前をマスクしない場合は、英大文字に変換されます。続く 2 つの数字は、結果をスライスで受け取る確率の指定です。次の 2 つの整数値は、スコアとヒット情報を要求するかどうかを指定します。最後の 2 つの値は出力値です。

パラメーターの詳細については、167 ページの『第 16 章 ストアード・プロシージャ検索関数』を参照してください。

注

より大きな結果セットを要求する場合は、ユーザー表スペースが必要です。使用できるものがない場合は、表スペースを作成してください。以下の例は、UNIX プラットフォーム上で表スペースを作成しています。

```
db2 "create user temporary tablespace tempts managed by system
      using ('/work/temptst.ts')"
```

分散 DB2 環境では、ストアード・プロシージャのために単一ノード上の管理表の表スペースを明示的に指定し、明示的にこのノードに接続しなければなりません。

SQL 表値関数を使用する検索

インターフェースをインプリメントしたもののすべての結果を戻す必要のない場合、あるいはストアード・プロシージャ・インターフェースを使用するために必要なすべての実メモリーがあるわけではない場合に、SQL 表値関数を使用してください。

使用可能な SQL 表値関数は 2 つありますが、いずれも `db2ext.textsearch` と呼ばれます。一方には、`db2ext.highlight` 関数と併用するための追加パラメーターがあります。詳しくは、82 ページの『HIGHLIGHT 関数の使用』を参照してください。

SQL 表値関数によって、結果の一部のみを入手する同じカーソル・インターフェースを使用できます。ただし、依然として、結果をユーザー表に結合する必要があります。これは、以下の例に示されています。

```
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"
```

SQL 表値関数から戻すことのできる値を以下に示します。

```
--> primKey <single primary key type>
the primary key
```

```
--> score          DOUBLE
the score value of the found document
```

```
--> NbResults      INTEGER
the total number of found results (same value for all rows)
```

```
--> numberOfMatches INTEGER
the number of hits in the document
```

注

使用できるのは単一の主キー列のみであることに注意してください。パラメーターの詳細については、159 ページの『DB2EXT.TEXTSEARCH』を参照してください。

HIGHLIGHT 関数の使用

使用可能な SQL 表値関数は 2 つありますが、いずれも `db2ext.textsearch` と呼ばれます。 `db2ext.highlight` 関数を使用するには、追加の `numberOfHits` および `hitInformation` の各パラメーターとともに `db2ext.textsearch` 関数を使用する必要があります。

この例では `db2ext.highlight` 関数を呼び出し、`db2ext.textsearch` 関数によって検出されたヒットをどれも強調表示することなく文書全体を表示します。

```
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
```

検索引き数は以下の結果を戻します。

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.

注

どの `db2ext.highlight` 関数の例でも、表関数 `db2ext.textsearch` は、「bestseller」、「peacekeeping」、または「attention」のいずれかの語を検索します。

この例では `db2ext.highlight` 関数を呼び出し、`db2ext.textsearch` 関数によって検出されたヒットをすべて強調表示して、文書全体を表示します。

```
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
```

検索引き数は以下の結果を戻します。

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.

この例では、db2ext.highlight 関数を呼び出して、最大で文書の 10 の部分 (ウィンドウ) を表示します。各ウィンドウ・サイズは 24 です。ヒットの両側のデータは約 12 バイトになります。さらに、表関数 db2.textsearch によって検出されたヒットが強調表示されます。

```
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
```

検索引き数は以下の結果を戻します。

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.

検出される最初のヒットは <bf>bestseller</bf> で、このヒットは最初のウィンドウを決定します。2 番目のヒットは <bf>peacekeeping</bf> で、最初のヒットから 8 バイトしか離れていないため、最初のウィンドウに完全に入れられます。3 番目のヒット <bf>soldiers</bf> は、最初のウィンドウの外側にあるので、新規ウィンドウを決定します。2 番目のヒット <bf>peacekeeping</bf> は <bf>soldiers</bf> ヒットの左側から 2 バイトしか離れていないので、これも 2 番目のウィンドウに入れられ、強調表示されます。4 番目のヒット <bf>attention</bf> は、2 番目のウィンドウの外側にあるので、新規ウィンドウを決定します。このウィンドウのサイズに以前のヒットまたは追加のヒットは入らないので、ヒットの周りのデータがウィンドウに入ります。

さらに、WINDOW_SEPARATOR が指定されていないため、デフォルトのウィンドウ区切り文字「...」が、3 つの文書ウィンドウを分離するために使用されます。

注

db2ext.highlight 関数を使用する際のハイパフォーマンスを確保するには、ユーザーは db2ext.textsearch 表値関数の検索結果を制限する必要があります。

パラメーターの詳細については、163 ページの『DB2EXT.HIGHLIGHT』を参照してください。

複数の列の検索

複数の列のテキスト索引を作成する必要がある場合、最も簡単なのは、SQL スカラー関数を使用し、それらの列の検索を結合する方法です。これを以下の例に示します。

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

表値関数の場合は、さらに困難です。パフォーマンス上の理由で和集合を使用しなければならない場合があるからです。表値関数の場合の別の方法は、ビューを使用し、ビューの列の中で表の列を組み合わせて、テキスト索引を作成するというものです。この方法を使うと、2つの別々のテキスト検索呼び出しを行う必要がなくなります。

テキスト列を組み合わせると、パフォーマンスが向上する可能性があります。とは言え、これは個々の検索上の要件に大きく依存します。

パフォーマンスに関する考慮事項

検索時のパフォーマンスを向上するために、次の点について考慮してください。

- SQL 内で検索する場合には、次の点について考慮してください。
 - パフォーマンスの低下に気付いた場合には、`explain` ステートメントを使用して、DB2 オプティマイザーの処理プランをチェックしてください。
 - パラメトリック検索を使用すると、検索が高速になることがあります。特に、結果のサイズを削減するために他の検索述部を使用している場合が当てはまります。
 - すべての結果を必要とするのでない場合、結果の制限キーワードを使用してください。
- ストアード・プロシージャを使用して検索する場合は、次の点について考慮してください。
 - 指定したキャッシュ表式は、データベースからメモリーにコピーされるため、ワークステーションに、このデータに使用できる十分なメモリーがあることを確認してください。十分メモリーがない場合は、ページング・スペースが使用され、検索パフォーマンスを低下させます。
- `NUMBEROFMATCHES` または `SCORE` 関数を使用する場合、`CONTAINS` 関数を併用しないと、照会パフォーマンスが低下することがあります。また、処理の重複を防ぐため、`CONTAINS` 関数のストリングが、`NUMBEROFMATCHES` または `SCORE` 関数に使用されているストリングと必ず正確に一致するようにしてください。

注

最新のパフォーマンスのヒントについては、DB2 Net Search Extender の Web サイト www.ibm.com/software/data/db2/extenders/netsearch/index.html を参照してください。

その他の検索構文の例

第 9 章 構造化文書を使用する作業

DB2 Net Search Extender を使用すると、構造化文書の表題、作成者、または記述などのテキスト・フィールドまたは数値フィールドに索引付けをしたり、検索ができるようになります。文書は XML、Outside-In、または HTML フォーマットで記述したり、ユーザー定義のタグ (GPP) を含めたりすることができます。

文書モデルのマークアップ・タグおよびフィールド名を使用して、文書内のどのフィールドに索引が付けられるか定義します。その結果、検索が使用可能になります。フィールドの名前 (セクション名とも呼ばれる) は、そのフィールドに対する照会で使用できます。

これらのフィールドで検索が可能になるためには、文書を含むテキスト索引を作成する際に `FORMAT AND MODEL FILE` を指定する必要があります。詳しくは、120 ページの『CREATE INDEX』を参照してください。

文書モデルの作成および定義の詳細については、171 ページの『第 17 章 構造化文書のサポート』を参照してください。

文書モデル構文については、239 ページの『付録 G. 文書モデルの参照情報』を参照してください。

第 10 章 検索項目を拡張するためのシソーラスの使用

特定の検索項目のみでなく、それに関連する項目も検索することによって、照会を拡張することができます。シソーラスからの関連検索項目の検索と抽出を行う Net Search Extender の関数を使用することによって、この処理を自動化できます。シソーラスとは、通常特定のサブジェクト・エリアに関する意味的に関連する用語の統制された語いです。

DB2 Net Search Extender を使用すれば、以前作成したシソーラスから追加の用語を追加することによって、検索項目を拡張することができます。照会でシソーラス拡張を使用する方法に関しては、147 ページの『第 14 章 検索引き数の構文』を参照してください。

シソーラスを作成して検索アプリケーションで使用するには、シソーラス定義ファイルが必要であり、これを内部フォーマット、シソーラス辞書にコンパイルする必要があります。

この章では、以下のことについて説明します。

- **シソーラスの構造**

シソーラスは、関係によって互いにリンクされたノードのネットワークのような構造になっています。このセクションでは、Net Search Extender の定義された関係およびユーザー独自の関係を定義する方法について説明します。

- **シソーラスの作成およびコンパイル**

ここでは、シソーラス定義ファイルの構文、およびシソーラス定義ファイルをシソーラス辞書にコンパイルするために使用するツールについて説明します。

シソーラスの構造

シソーラスは、関係によって互いにリンクされたノードのネットワークのような構造になっています。Net Search Extender は、シソーラスで特定の用語の検索から始め、次に、用語間の関係のパスに従い、そのプロセスで見つかった用語を出力します。

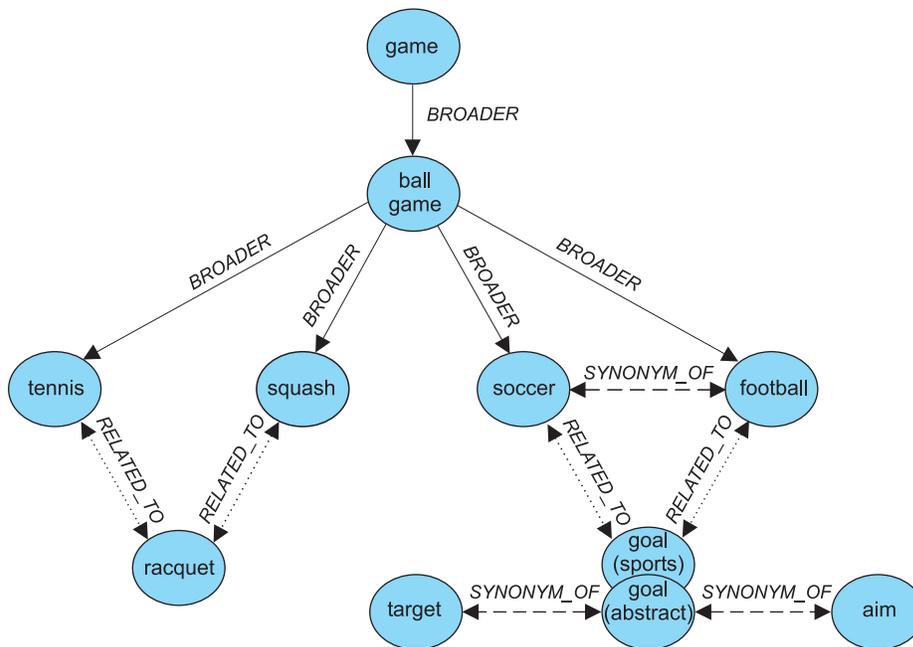


図 28. シソーラスの構造の例

シソーラスの項目は関係によって結び付けられています。BROADER などの関係名を使用すると、関係階層での名前の付いた特定の線に拡張を制限できます。関係には双方向のものと単一方向のものがあります。たとえば、BROADER は単一方向関係の名前です。

定義済みシソーラス関係

Net Search Extender で事前に定義された関係を以下に示します。

- 連想関係

連想関係は、同じ概念を表現しているわけではないものの、相互に関係する 2 つの用語の間の双方向の関係です。

定義済み連想関係: RELATED_TO

例:

```
tennis RELATED_TO racket
football RELATED_TO goal (sports)
```

- 同義語関係

同義語関係は、同じ、または類似した意味を持ち、相互に代替語として使用できる 2 つの用語の間の双方向の関係です。この関係は、たとえば、用語とその省略形などに使用できます。

定義済み同義語関係: SYNONYM_OF

例:

```
spot SYNONYM_OF stain
US SYNONYM_OF United States
```

90 ページの図 28 では、同じシソーラスで、2 つの goal という用語が示されています。1 つはコメント (sports) が付いており、もう 1 つはコメント (abstract) が付いています。用語が同じスペルを持っていても、同義語関係で異なるワード・グループを接続できます。シソーラスを定義する際に、異なる関係を使用することによってこれをモデル化することができます。

• 階層関係

階層関係は 2 つの用語間の単一方向の関係です。ある用語は別の用語より広い (よりグローバルな) 意味をもっています。この関係を使用すると、方向により、より限定的な用語またはよりグローバルな用語を検索できます。

定義済み階層関係:

- NARROWER 幅が狭くなる関係をモデル化します。

NARROWER 関係は、より限定的用語のシーケンスをモデル化するためのものです。幅が狭くなる関係の先へいくほど、用語がより具体的なものになっていきます。たとえば、NARROWER 関係に沿って house という用語を検索すると、結果は限定化の度合いが増していく用語リストの skyscraper、palace、church、chapel、cathedral などになる可能性があります。

- BROADER 幅の広がる関係をモデル化します。

BROADER 関係は、だんだんグローバルになっていく用語のシーケンスをモデル化するためのものです。このような関係の先へいくほど、用語がより具体的でなくなっていきます。たとえば、BROADER 関係に沿って、house という用語を検索すると、結果は、グローバル化の度合いが増していく用語リストの、building、construction、object などになる可能性があります。

ユーザー独自の関係の定義

Net Search Extender を使用すると、ユーザー独自の RELATED_TO、NARROWER、および BROADER のシソーラス関係を定義できます。各関係名はユニークなものでなければならぬため、RELATED_TO(42) などのようにユニーク番号を追加することによって、そのような関係名を限定する必要があります。

NARROWER(42) などのように、同じリレーションシップ番号を使用して異なるタイプのリレーションシップを定義できます。番号 0 は、Net Search Extender の定義済み関係を示すために使用されます。

シソーラスの作成およびコンパイル

Net Search Extender 関数で使用できるシソーラスを作成するには、以下のステップを使用します。

1. シソーラス定義ファイルを作成します。
2. 定義ファイルをシソーラス辞書にコンパイルします。

シソーラス定義ファイルの作成

ユーザー独自のシソーラスを作成するには、まず、テキスト・エディターを使用して、定義ファイルで内容を定義します。

シソーラスを使用する作業

制約事項 拡張子を含むファイル名の長さは、256 文字を超えてはなりません。同じディレクトリーに複数のシソーラスをもつことができますが、各シソーラスごとに別々のディレクトリーをもつことをお勧めします。

サンプルの英語のシソーラス定義ファイル `nsesamplethes.def` が提供されています。Windows システムのシソーラス・ディレクトリーは次のとおりです。

```
<sql1lib>\db2ext\thes
```

UNIX システムでの、シソーラス・ディレクトリーは次のとおりです。

```
<instance_owner_home>/sql1lib/db2ext/thes
```

定義ファイルの最初のいくつかの定義グループを以下に示します。

```
: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
:
:
:
```

図 29. サンプル・シソーラス定義ファイルの一部

各定義グループの構文については、187 ページの『第 18 章 シソーラス・サポート』を参照してください。

各メンバーは、単一行に書き込む必要があります。各関連用語は、関係名の前に置く必要があります。メンバー用語が互いに関係している場合、メンバー関係を指定します。

メンバー用語と関連用語の長さは 64 文字に制限されています。同一文字の 1 バイト文字と 2 バイト文字は、同じと見なされます。大文字と小文字は区別されません。用語には空白文字を含めることができ、1 バイト文字のピリオド "." またはコロン ":" を使用できます。

ユーザー定義の関係は、すべて連想タイプに基づいています。これらは、1 から 128 の間のユニークな番号によって識別されます。

定義ファイルのシソーラス辞書へのコンパイル

シソーラス定義ファイルをコンパイルするには、`db2extth` を実行します。コマンド構文については、139 ページの『DB2EXTTH (ユーティリティ)』を参照してください。

パーティション化された環境でシソーラス辞書を使用するには、作成されたファイルにすべての物理ノードがアクセスできることを確認してください。

ヒント

263 ページの『付録 M. シソーラス・ツールが戻すメッセージ』を参照してください。

シソーラスを使用する作業

第 2 部 参照情報

第 11 章 インスタンス所有者用の管理コマンド

この章では、インスタンス所有者用の管理コマンドの構文について説明しています。インスタンス所有者の管理は、DB2 Net Search Extender の状況チェック、ロッキングおよび更新サービス、さらにこれらのサービスの開始および停止から構成されています。

詳細に関しては、27 ページの『第 5 章 Net Search Extender インスタンス・サービス』を参照してください。

コマンドは DB2TEXT コマンドのバリエーションであり、DB2 インスタンスに特有の DB2 Net Search Extender サービスの管理を行うことができます。

コマンド	用途	ページ
CONTROL	フル・テキストの索引ロックをリストし、削除する。キャッシュ状態もリストする。	98
START	DB2 Net Search Extender インスタンス・サービスを開始する。	100
STOP	DB2 Net Search Extender インスタンス・サービスを停止する。	101

CONTROL

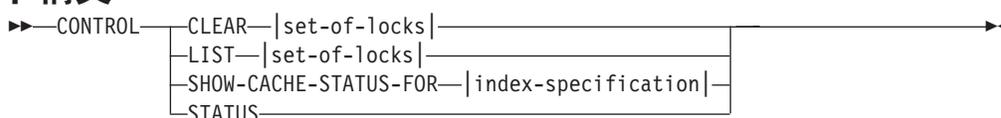
このコマンドを使用すれば、Net Search Extender インスタンス・サービスによって管理されるフル・テキストの索引ロックをリストしたり削除することができます。ロッキング・サービスおよび更新サービスが実行中の場合は、その状況、およびアクティブ化されたキャッシュに関する情報を表示できます。

分散 DB2 環境では、現行パーティションにのみ影響を及ぼします。ユーザーには、必要なパーティションに対して DB2 コマンド、`db2_all` を呼び出す責任があります。

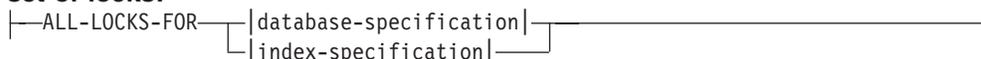
許可

このコマンドをサーバー上の DB2 インスタンス所有者として実行する必要があります。

コマンド構文



set-of-locks:



index-specification:



Database-specification:



コマンド・パラメーター

CLEAR

`CLEAR` を使用して一連のロックを強制的にクリーンアップします。ロックに関する問題の原因を確認した後で、このコマンドを慎重に使用してください。

LIST

特定の索引またはデータベース用に保持されている現行のロックについて、情報を得るには、`LIST` を使用します。更新ロックが存在する場合、処理された文書に関する情報を入手できます。

更新処理中に更新された文書の数を表示するために `LIST` を使用することもできます。これは、索引にロックが保持されている場合にだけあてはまることに注意してください。

レプリケーション・キャプチャー表を使用する場合、更新操作はありません。その代わりに、挿入操作は、索引が作成されたソース表での挿入または更新操作のいずれかによるものである場合があります。

set-of-locks

指定されたデータベースまたは索引のロックについてのみ処理を行います。

SHOW CACHE STATUS FOR

指定された索引のキャッシュ表のアクティブ化状況を表示します。これは、「アクティブ化されていない (Not Activated)」または「現在アクティブ化されている (Currently Activated)」のいずれかになります。キャッシュがアクティブ化されている場合、キャッシュ・メモリーの使用に関する詳細が表示されます。たとえば、最大キャッシュ・サイズ (メガバイト)、挿入できる文書の最大数、およびキャッシュ表に残っているスペース (キロバイト) などについてです。

STATUS

STATUS キーワードを使用することによって、このコマンドは Net Search Extender インスタンス・サービスのロッキング・サービスおよび更新サービスが稼働中であるかどうかを表示します。

DATABASE database name

使用中のサーバー上のデータベースの名前。

INDEX index-schema.index-name

現在、使用中のテキスト索引のスキーマおよび名前。これは CREATE INDEX コマンドで指定されます。

使用法

管理コマンド・エラー・メッセージがロックに関する問題の発生を示している場合、競合するタスクが実行中でないことを確認してください。たとえば、UPDATE コマンドの実行中に ALTER コマンドを実行しようとしていないか、などです。そして、索引のロックをすべて解除してください。

インクリメンタル索引更新に SHOW CACHE STATUS FOR を使用して、指定されたメモリー・サイズが、次の更新時にも引き続き更新情報をすべて収容できるだけの大きさを持っているか、アクティブ化が実行済みであるかどうかをチェックしてください。

START

このコマンドは、フル・テキスト索引のロックング、および DB2 サーバー上のフル・テキスト索引の自動更新を制御するデーモンを開始します。

注

コマンドは、索引の一時キャッシュ表をまったくアクティブ化しないので、ストアド・プロシージャーによる検索を行うには、個別の ACTIVATE CACHE コマンドが必要です。

許可

このコマンドをサーバー上または、分散 DB2 環境の任意のサーバーの DB2 インスタンス所有者として実行する必要があります。

コマンド構文

▶—START—▶

コマンド・パラメーター

なし。

使用法

Windows では、コマンドはサービス db2ext-<InstanceName> を開始します。このコマンドは、通常の Windows 方式でも開始できます。

フルテキスト索引をロックングする場合、要件を満たすように構成ファイルを変更できます。詳しくは、27 ページの『ロックング・サービス』を参照してください。

STOP

このコマンドは、Net Search Extender のロッキング・サービスおよび更新サービスを停止します。

許可

このコマンドをサーバー上または、分散 DB2 環境の任意のサーバーの DB2 インスタンス所有者として実行する必要があります。

コマンド構文

```
» STOP [FORCE]
```

コマンド・パラメーター

FORCE

処理がロックを保留中であつたり、キャッシュ表が任意の索引に対してアクティブ化されている場合であっても、サービスを停止します。FORCE を指定しないと、コマンドはこれらのケースで失敗します。

使用法

Net Search Extender インスタンス・サービスを停止すると、特定の Net Search Extender コマンドをそれ以上使用できなくなります。以前、アクティブ化されたキャッシュを索引に対して使用した場合は、サービスを再始動するときに、一時キャッシュを再びアクティブ化する必要があります。

アクティブ化されたキャッシュまたは実行中の Net Search Extender コマンドは、サービスを停止しないことに注意してください。

フルテキスト索引をロッキングする場合、要件を満たすように構成ファイルを変更できます。詳しくは、27 ページの『ロッキング・サービス』を参照してください。

STOP コマンド

第 12 章 データベース管理者用の管理コマンド

この章では、データベース管理者用の管理コマンドの構文について説明しています。データベース管理コマンドには、DB2 Net Search Extender が使用するためのデータベースのセットアップ、およびこのセットアップを使用できないようにする作業が含まれています。

31 ページの『第 6 章 テキスト索引の作成と保守』は、これらのコマンドの使用方法について説明しています。

ENABLE DATABASE および DISABLE DATABASE コマンドのみは、DB2TEXT コマンドのバリエーションですが、次のすべてのコマンドは、データベース・レベル上の管理を可能にします。

コマンド	用途	ページ
ENABLE DATABASE	現行データベースがフル・テキスト索引を作成できるようにする。	104
DISABLE DATABASE	データベース用に DB2 Net Search Extender によって完了した準備作業をリセットする。	106
DB2EXTDL (ユーティリティー)	データ・リンク・テキスト列の内容を検索するデフォルトの UDF。	108
DB2EXTHL (ユーティリティー)	デフォルトの UDF は、100 KB の文書を受け取って、200 KB の CLOB を戻す。	109

ヒント

db2text コマンドの一部としてデータベース接続が指定されていない場合、db2text 実行可能コードは、環境変数 DB2DBDFT で指定されたデフォルトのデータベースに対する暗黙接続を行います。

ENABLE DATABASE

このコマンドを実行することによって、データベースはテキスト列上にフル・テキスト索引を作成して活用できるようになります。

許可

データベース管理者として、データベースを使用可能にするためにこのコマンドを実行する必要があります。このためには、DBADM を DB2 インスタンス所有者に付与できる SYSADM 権限が必要です。

コマンド構文

```

▶▶ ENABLE-DATABASE-FOR-TEXT |-----▶▶
    |-----|
    |-----|connection-options|-----|
  
```

connection-options:

```

|-----|
|-----|CONNECT-TO database-name |-----|
|-----|USER userid USING password |-----|
  
```

コマンド・パラメーター

CONNECT TO database-name

このコマンドのターゲットであるデータベースの名前。DB2DBDFT が設定され、必要な DB2 権限を持つユーザー ID の下でユーザーがコマンドを実行している場合は、このパラメーターを省略できます。

USER userid USING password

password および userid を使用してデータベースに接続します。

使用法

このコマンドは、接続されたデータベースを DB2 Net Search Extender で使用するために準備します。これは必須のステップであり、このステップを実行しないと、データベース内の表/列に DB2 Net Search Extender 索引を作成できるようになりません。

コマンド実行後に確立されるデータベース・デフォルトは、DB2EXT.DBDEFAULTS カタログ・ビューを使用して表示できます。

データベースへの変更

このコマンドは、使用可能になったデータベースの DB2 インスタンスに関連した DB2 インスタンス所有者に DBADM 権限を付与します。

ENABLE DATABASE コマンドは、DB2 Net Search Extender カタログ、UDF およびストアード・プロシージャなどの各種のデータベース・オブジェクトをスキーマ DB2EXT に作成します。コマンド実行後に、次のカタログ・ビューが使用できます。

```

db2ext.dbdefaults
db2ext.textindexes
db2ext.textindexformats
db2ext.indexconfiguration
db2ext.proxyinformation
  
```

DB2 テキスト情報エクステンダー・ビューも、逆方向互換性の理由により、使用できることに注意してください。詳しくは、199 ページの『付録 C. Net Search Extender インフォメーション・カタログ』を参照してください。

さらに、前記の表は、データベースのデフォルトの表スペース (IBMDEFAULTGROUP) に配置されることに注意してください。これは、db2nodes.cfg で定義されたノード全域に配布されます。

ファイル・システムへの変更

なし。

注

データベースにテキスト索引が定義されている場合に、そのデータベースを使用不可にしようとする、失敗します。これらの索引を 1 つ 1 つ除去した上で、何か問題が生じるかどうかを検査するようお勧めします。 `disable database for text force` コマンドを使用した場合に保証されるのは、データベース内の Net Search Extender カタログ表が削除されるということに過ぎません。

しかし、完全にドロップできない索引があった場合は、手作業でクリーンアップしなければならないリソースがまだあるかもしれません。以下のものが含まれます。

- 索引ディレクトリー、作業ディレクトリー、およびキャッシュ・ディレクトリー内のファイル
- `ctedem.dat` 内のスケジューラー項目
- レプリケーション・キャプチャー・オプションを使用して索引を作成した場合には、リモート・データベースの表にある、`IBMSNAP_SIGNAL`、`IBMSNAP_PRUNE_SET`、および `IBMSNAP_PRUNCNTL` の各項目を手作業で削除しなければなりません。これらの項目は、`APPLY_QUAL="NSE" || <instance name> and TARGET_SERVER= <database name>` コマンドで簡単に識別できます。

次の例では、インスタンスは DB2、データベースは SAMPLE です。

```
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';
```

データベースへの変更

DB2 Net Search Extender を使用可能にする際にデータベースに対して行われる以下の変更は削除されます。

- データベース内での DB2 Net Search Extender カタログ・ビュー。
- DB2 Net Search Extender によって作成されるすべてのデータベース・オブジェクト。

ファイル・システムおよび共有メモリーへの変更

`FORCE` オプションを使用する場合、索引ファイルは削除されます。

`FORCE` オプションを使用する場合、索引のアクティブ化されたキャッシュに関するキャッシュが削除されます。詳しくは、137 ページの『`DROP INDEX`』を参照してください。

DB2EXTDL (ユーティリティー)

デフォルトでは、データ・リンク・テキスト列の内容を検索する UDF は、100 KB BLOB を戻します。データ・リンクが参照するデータベースで最大の文書のサイズに応じて、この値を増減させることができます。

許可

データベース管理者として、データベースを使用可能にするためにこのコマンドを実行する必要があります。このためには、DBADM を DB2 インスタンス所有者に付与できる SYSADM 権限が必要です。

コマンド構文

▶▶—db2extdl—*new-result-size*————▶▶

コマンド・パラメーター

new-result-size

データ・リンクの内容を検索する UDF の新規の結果サイズ (K バイト)。
これは <2097152 の正の整数です。

DB2EXTHL (ユーティリティー)

デフォルトでは、強調表示 UDF は入力として最大 100 KB のサイズの文書を受け取り、200 KB の CLOB を戻します。データベースにある最大の文書のサイズに応じて、入力値を大きくすることができます。最大サイズは 1 GB です。

許可

データベース管理者として、データベースを使用可能にするためにこのコマンドを実行する必要があります。このためには、DBADM を DB2 インスタンス所有者に付与できる SYSADM 権限が必要です。

コマンド構文

▶▶—db2exthl—*new-highlight-input-size*————▶▶

コマンド・パラメーター

new-highlight-input-size

強調表示 UDF の新しい結果サイズ (K バイト)。これは <1048576 の正の整数です。

DB2EXTHL (ユーティリティー) コマンド

第 13 章 テキスト表所有者用の管理コマンド

この章では、テキスト表所有者用の管理コマンドの構文について説明しています。

31 ページの『第 6 章 テキスト索引の作成と保守』は、これらのコマンドの使用方法について説明しています。

これらのコマンドは DB2TEXT コマンドのバリエーションです。このコマンドを使用すると、表の所有者は、表の列についてのフル・テキスト索引を作成し、それを操作できます。

コマンド	用途	ページ
ACTIVATE CACHE	キャッシュをアクティブ化する。この結果、ストアード・プロシージャを使用した検索が可能になる	112
ALTER INDEX	索引の特性を変更する	114
CLEAR EVENTS	索引更新時に使用される索引イベント表から索引イベントを削除する	118
CREATE INDEX	フル・テキスト索引を作成する	120
DEACTIVATE CACHE	キャッシュを非アクティブ化する。この結果、ストアード・プロシージャを使用した検索が実行できなくなる	135
DB2EXTTH (ユーティリティー)	シソーラス定義ファイルをコンパイルする	139
DROP INDEX	テキスト列用のフル・テキスト索引をドロップする	137
ENABLE DATABASE	現行データベースがフル・テキスト索引を作成できるようにする	104
UPDATE INDEX	テキスト列の現行の内容に基づいて索引付け処理を開始する	141
HELP	DB2TEXT コマンド・オプションのリストを表示する	145
COPYRIGHT	Net Search Extender の製品および著作権情報を表示する	146

ヒント

db2text コマンドの一部としてデータベース接続が指定されていない場合、db2text 実行可能コードは、環境変数 DB2DBDFT で指定されたデフォルトのデータベースに対する暗黙接続を行います。

ACTIVATE CACHE

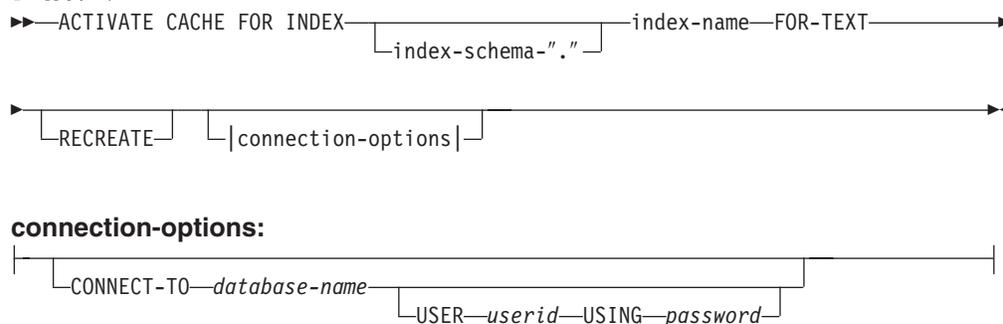
このコマンドは、DB2 ユーザー表または永続キャッシュのいずれかからキャッシュ済みの表をアクティブ化します。完了後は、ストアード・プロシージャを使用した検索操作が可能です。詳しくは、167 ページの『第 16 章 ストアード・プロシージャ検索関数』を参照してください。

このコマンドは、索引が CACHE TABLE オプションを指定して作成された場合にのみ使用可能であることにも注意してください。詳しくは、120 ページの『CREATE INDEX』を参照してください。

許可

DB2 カタログ・ビューによると、このコマンドのユーザー ID は、フル・テキスト索引が作成された表に対して CONTROL 特権が必要です。

コマンド構文



コマンド・パラメーター

index-schema

テキスト索引のスキーマ (CREATE INDEX コマンドで指定されたもの)。スキーマが指定されていない場合は、DB2 接続のユーザー ID が使用されます。

index-name

テキスト索引の名前 (CREATE INDEX コマンドで指定されたもの)。

RECREATE

永続キャッシュを使用する索引にのみ適用されます。既存のキャッシュは削除されます。アクティブ化なしで更新が完了した場合は、永続キャッシュはデータベースから自動的に再構成されます。

CONNECT TO database-name

このコマンドのターゲットであるデータベースの名前。DB2DBDFT が設定されており、ユーザーがサーバーでコマンドを実行している場合は、このパラメーターを省略できます。ユーザー ID には必須 DB2 許可がなければならぬことに注意してください。

USER userid USING password

password および userid を使用してデータベースに接続します。指定しない場合は、パスワードなしで現行ユーザー ID から接続が試行されます。

使用法

次のコマンドのいずれかが索引上で実行中の場合には、このコマンドを発行できないことに注意してください。

- UPDATE INDEX
- ALTER INDEX
- DROP INDEX
- CLEAR EVENTS
- DEACTIVATE CACHE

注

永続キャッシュが使用された場合でも、キャッシュ済み表のアクティブ化の際に、キャッシュ表を最初から再び作成しなければならない可能性があります。これは、永続キャッシュが非アクティブ化されている間に、更新操作が実行された場合に発生します。

キャッシュの構築に必要なメモリーの量は、現行の文書数と結果の列のサイズから、動的に計算されます。PCTFREE 値を使用して、 $100/(100-PCTFREE)$ の係数で計算されたメモリーの最少量を増やします。PCTFREE 値は CREATE または ALTER INDEX コマンドに指定される値です。

したがって、PCTFREE は、キャッシュがアクティブ化されている場合に挿入操作に確保された、割り振り済みキャッシュのパーセンテージを表します。ACTIVATE CACHE コマンドを実行するたびに、実際のメモリー・サイズが再評価されるため注意してください。

ファイル・システムへの変更

永続キャッシュをインプリメントするためのファイルが作成されます。

コマンド・パラメーター

index-schema

CREATE INDEX コマンドで指定されたテキスト索引のスキーマ。スキーマが指定されていない場合は、DB2 接続のユーザー ID が使用されます。

index-name

CREATE INDEX コマンドで指定されたテキスト索引の名前。

INDEX DIRECTORY directory

テキスト索引の保管先ディレクトリー・パス。ディレクトリーは索引データを含むので、ディレクトリーには、DB2 インスタンス所有者ユーザー ID 用の読み取り/書き込み、および実行権限があることを確認してください。

分散 DB2 環境では、このディレクトリーはすべてのノード上に存在する必要があることに注意してください。サブディレクトリー NODE<nr> は、サーバーの論理ノード上の索引を区別するために、このディレクトリーの下に作成されます。以前の索引ディレクトリーからの索引ファイルは、すべて削除されます。

WORK DIRECTORY workdirectory

検索操作と管理操作中に一時ファイルを保管します。新規索引ディレクトリーとは無関係に、別個の作業ディレクトリーを変更できます。

ディレクトリーが存在しない場合は、DB2 インスタンス所有者ユーザー ID 用に作成されます。ディレクトリーが存在する場合は、ディレクトリーがインスタンス所有者用に UNIX プラットフォーム上での読み取り/書き込み権限を持っていることを確認してください。

分散 DB2 環境では、このディレクトリーはすべてのノード上に存在する必要があることに注意してください。サブディレクトリー NODE<nr> は、サーバーの論理ノード上の索引を区別するために、このディレクトリーの下に作成されます。以前の索引ディレクトリーからの一時索引ファイルは、すべて削除されます。

CACHE TABLE PERSISTENT IN directory

非アクティブ化またはシステム・リポート後も CREATE INDEX のキャッシュに入れられた表が永続的であることを指定します。いずれの場合でも、迅速な ACTIVATE CACHE の実行が可能になります。永続キャッシュは、指定されたディレクトリーに保管されます。

以前に作成された永続キャッシュは、新しいロケーションに移されます。このロケーションには、常に非アクティブ化された索引が必要です。

CACHE TABLE TEMPORARY

キャッシュ結果表が、現在一時的であり、それまで存在した永続キャッシュが削除されたことを指定します。この変更は、非アクティブ化された索引が必要であることに注意してください。

MAXIMUM CACHE SIZE memsize

ACTIVATE CACHE 時に作成されるキャッシュ済み表の新たな最大サイズを指定します。memsize パラメーターは、メガバイト単位で正の整数として指定します。

整数が小さすぎると、ACTIVATE CACHE コマンドは失敗します。実際のキャッシュ・サイズは ACTIVATE CACHE コマンド時に計算されます。この変更には、非アクティブ化された索引が必要です。

PCTFREE percentage

追加の文書のためにフリーにしておくキャッシュのパーセンテージを指定します。パーセンテージは 0 以上で 100 より小さい整数値でなければなりません。それまでの永続キャッシュは削除され、この変更には非アクティブ化された索引が必要であることに注意してください。 112 ページの『ACTIVATE CACHE』を参照してください。

UPDATE FREQUENCY

以下のパラメーターを使用すると、索引更新の頻度は更新が実行される時期を決定します。

- **D** 索引が更新される曜日: * (毎日) または 0.6 (0=日曜日)
- **H** 索引が更新される時間: * (毎時) または 0.23
- **M** 索引が更新される分: 0.59
- **NONE** これ以上の索引更新はありません。これは、これ以上の変更がないテキスト列用に意図されたものです。

UPDATE FREQUENCY キーワードを指定しない場合、頻度設定は未変更のままです。

UPDATE MINIMUM minchanges

索引が増大されて更新されるまでに、テキスト文書に許可される最小の変更の数。 UPDATE MINIMUM キーワードを指定しないと、設定は変更されません。

RECREATE ON UPDATE オプションを使用して索引を作成しなかった場合は、UPDATE MINIMUM しか変更できないことに注意してください。

COMMITCOUNT FOR UPDATE count

更新処理の場合、コミット・カウントを指定できます。詳しくは、141 ページの『UPDATE INDEX』を参照してください。これは、UPDATE コマンド、および更新処理をスケジュールする UPDATE FREQUENCY 指定の両方に適用されます。

RECREATE ON UPDATE オプションを使用して索引を作成しなかった場合は、COMMITCOUNT しか変更できないことに注意してください。

また、索引に REPLICATION 文節を指定して作成した場合は、COMMITCOUNT を変更できないことにも注意してください。

CONNECT TO database-name

このコマンドのターゲットであるデータベースの名前。DB2DBDFT が設定されており、ユーザーがサーバーでコマンドを実行している場合は、このパラメーターを省略できます。ユーザー ID には必須 DB2 許可がなければならないことに注意してください。

USER userid USING password

password および userid を使用してデータベースに接続します。指定しない場合は、パスワードなしで現行ユーザー ID から接続が試行されます。

使用法

次のコマンドのいずれかが索引上で実行中の場合には、このコマンドを発行できないことに注意してください。

- ALTER INDEX
- CLEAR EVENTS
- ACTIVATE CACHE
- DROP INDEX
- UPDATE INDEX
- DEACTIVATE CACHE

キャッシュ・オプションを指定して索引を作成する場合、索引がアクティブになっているときは、索引ディレクトリーに対して ALTER INDEX コマンドを使用できません。まず始めに、キャッシュを非アクティブにします。

分散 DB2 環境では、キャッシュ・オプション付きのテキスト索引は、単一ノード表スペース上にのみ許可されています。

データベースへの変更

DB2 Net Search Extender カタログ・ビューを変更します。

ファイル・システムへの変更

- 索引ディレクトリーおよび作業ディレクトリーに NODE<nr> サブディレクトリーを作成します。
- 索引ファイルを移動します。
- 永続キャッシュ・ディレクトリーを作成します。
- 永続キャッシュ・ファイルを移動します。

特に文書の索引を再作成する場合は、表のテキスト列の内容と索引の整合性を確保するようにしてください。

次のコマンドのいずれかが索引上で実行中の場合には、このコマンドを発行できないことに注意してください。

- UPDATE INDEX
- ALTER INDEX
- ACTIVATE CACHE
- DEACTIVATE CACHE
- DROP INDEX

CREATE INDEX

このコマンドは、DB2 Net Search Extender フル・テキスト照会で使用するために、テキスト列上にフル・テキスト索引を作成します。

分散 DB2 環境では、フル・テキスト索引は、ユーザー表が定義された表スペースのすべてのパーティションに作成されます。表スペースの配布に対する以降の変更は許可されておらず、管理コマンドおよび検索処理において、予期しない動作につながります。

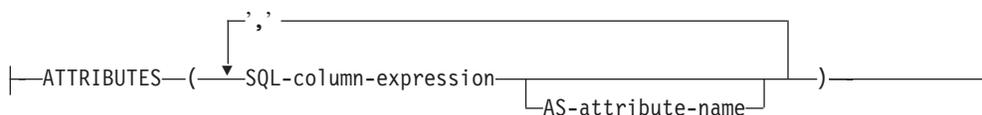
許可

DB2 カタログ・ビューによると、このコマンドのユーザー ID は、フル・テキスト索引が作成された表に対して CONTROL 特権が必要です。

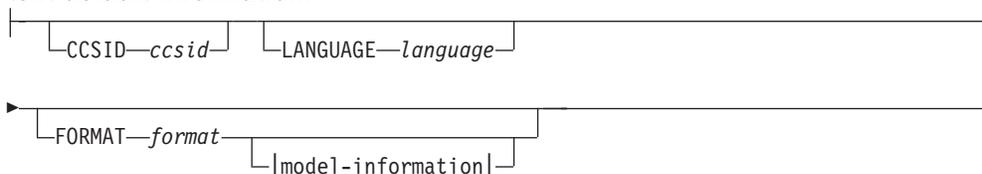
コマンド構文



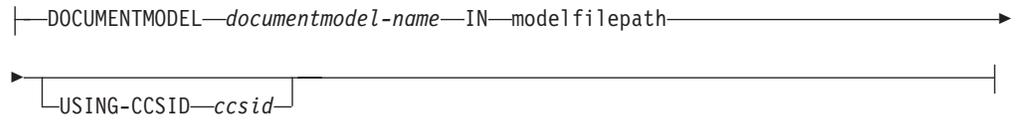
attribute-list:



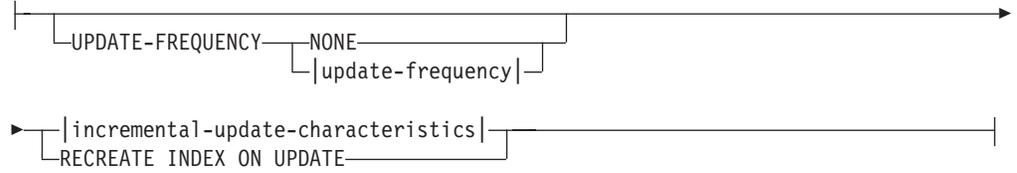
text-default-information:



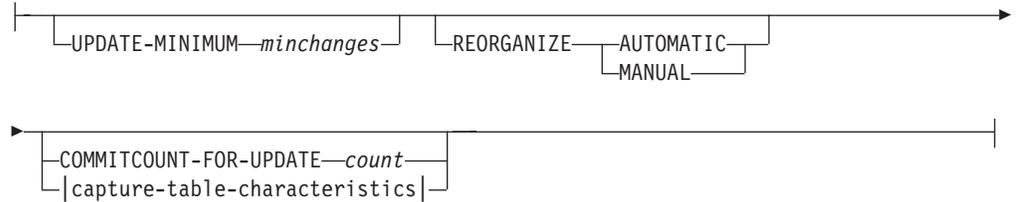
model-information:



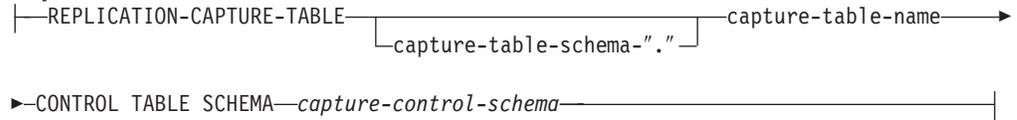
update-characteristics:



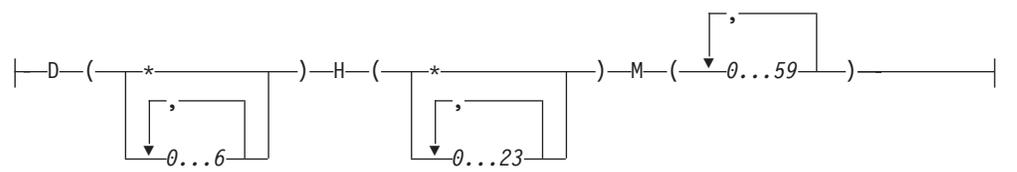
incremental-update-characteristics:



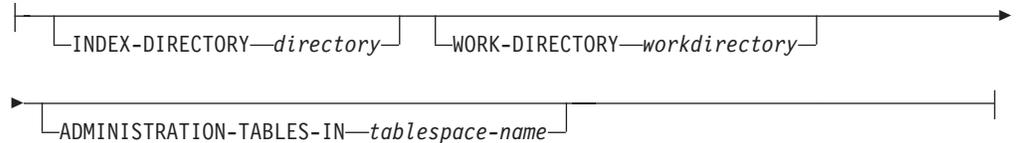
capture-table-characteristics:



update-frequency:



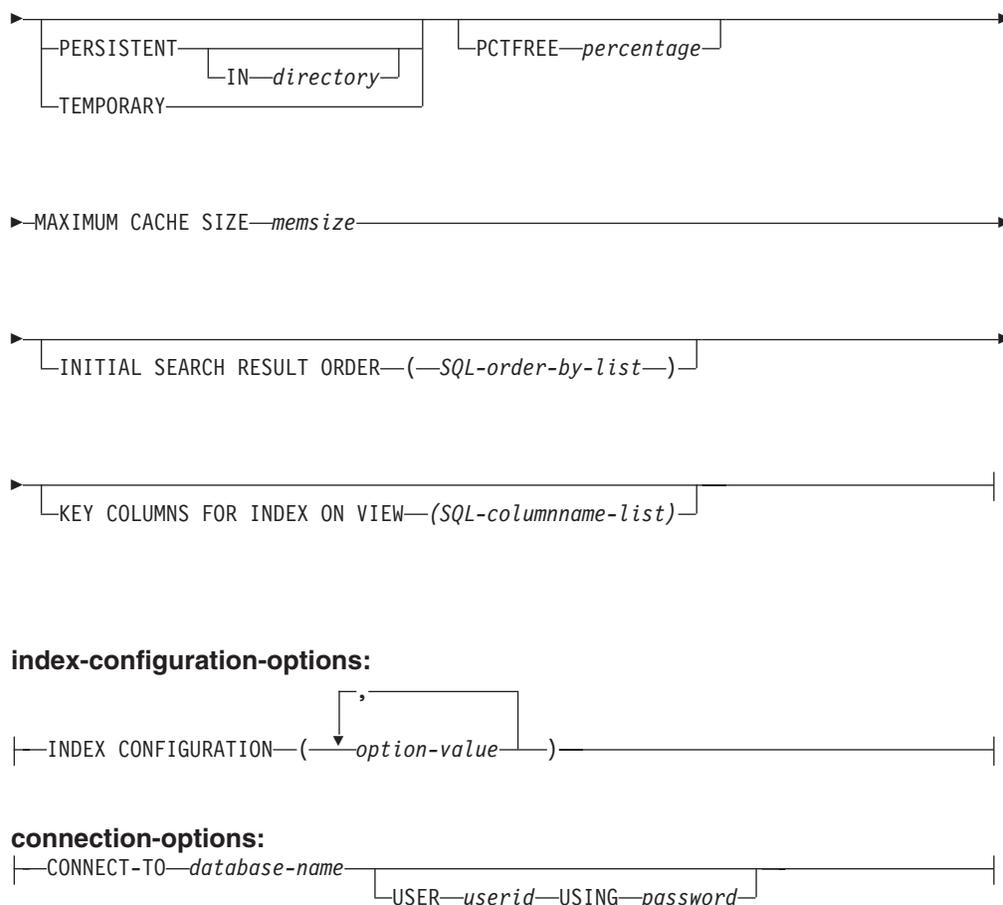
storage-options:



cache-search-results-options:



CREATE INDEX コマンド



コマンド・パラメーター

index schema

テキスト索引のスキーマ。索引特定の管理表の DB2 スキーマ名として使用します。スキーマが指定されていない場合は、DB2 接続のユーザー ID が使用されます。索引スキーマは、有効な DB2 スキーマ名でなければならないことに注意してください。

index name

索引の名前。索引スキーマとともに、この名前はデータベースのフル・テキスト索引を一意に識別します。また、索引イベント表の名前としても機能します。

詳細については 199 ページの『付録 C. Net Search Extender インフォメーション・カタログ』を参照してください。索引名は、有効な DB2 索引名でなければならないことに注意してください。

table schema

索引が作成される表、ニックネーム、またはビューのスキーマ。スキーマが指定されていない場合は、DB2 接続のユーザー ID が使用されます。

table name

フル・テキスト索引が作成される列を含む、接続されたデータベース内のテキスト表、ニックネーム、またはビューの名前。

表名が DB2 基本表を参照しない場合は、次の制限があることに注意してください。

- ビューは、ストアド・プロシージャ検索および表値関数検索のみを許可します。したがって、索引またはビューのキー列を `KEY COLUMNS FOR INDEX ON VIEW` 文節を使用して指定する必要があります。
- キャプチャー表を使用しないニックネームへのインクリメンタル索引更新の場合、ログ表が作成されます。ニックネーム表またはビューのデータに変更がある場合は、手動でログ表に入力する必要があります。基本表の場合、この作業は自動的に実行されるので、ユーザーはログ表に触れては**なりません**。ログ表のレイアウトについては、199 ページの『付録 C. Net Search Extender インフォメーション・カタログ』を参照してください。
- DB2 の述部 `CONTAINS`、`SCORE`、および `NUMBEROFMATCHES` は、基本表またはニックネーム上の索引にのみ許可されており、ビュー上の索引には許可されていません。
- キャッシュ検索結果オプションをコマンドに指定した場合にのみ、ビュー上の索引は許可されます。

text-column-name

フル・テキスト索引の作成に使用されるテキストを含む列の名前。列は以下のいずれかのタイプである必要があります。

- CHAR (FOR BIT DATA)
- VARCHAR (FOR BIT DATA)
- LONG VARCHAR (FOR BIT DATA)
- CLOB
- DBCLOB
- BLOB
- GRAPHIC
- VARGRAPHIC
- LONG VARGRAPHIC
- DATALINK

列タイプがこれらのどれにも当てはまらない場合は、

function-schema.function-name を使用して、トランスフォーメーション関数を指定し、列タイプを変換します。

データ・リンク列を使用する場合、参照される内容が索引付けのために取り出されることに注意してください。この処理はデータ・リンク値の一部であるプロトコルを経由します (例: Http)。`"file"` または `"unc"` 以外のプロトコルを使用する場合は、データ・リンク値の一部であるサーバーによって、これらがサポートされていることを確認してください。ファイル内容を取得するためにプロキシ・サーバーが必要になることもあるので、データベース管理者は、索引を作成する前に `DB2EXT.PROXYINFORMATION` 表でプロキシ・サーバーを指定できます。

同じ列に複数の索引が許可されていますが、これは次の条件を満たす場合**だけである**ことに注意してください。

索引がビュー上に作成されている

したがって、索引は CONTAINS、SCORE または NUMBEROFMATCHES 検索引き数では使用できません。

索引が表上に作成されている

作成されたすべての索引が同期されている場合には、CREATE INDEX コマンドで指定される以下の内容について、同じ列のプロパティが等しくなります。

- 関数名およびスキーマ
- ATTRIBUTES
- CCSID
- LANGUAGE
- FORMAT
- DOCUMENTMODEL
- INDEX CONFIGURATION

この場合、CONTAINS、SCORE または NUMBEROFMATCHES 引き数によって選択された索引は問題なく使用することができます。

function-schema.function-name

サポートされないタイプの列にあるテキスト文書にアクセスするために使用されるユーザー定義関数のスキーマおよび名前。関数は、任意の列タイプの 1 つの入力パラメーターを使用して、列タイプ変換を実行します。上記の Net Search Extender サポート・タイプの 1 つの値を戻します。

ATTRIBUTES (SQL-column-expression AS Attribute-name, ...)

列式の内容がテキスト列に加えて索引付けされることを決定します。この内容は検索ステートメントの ATTRIBUTE 文節によって検索することも可能です。SQL 列式は、索引が作成された表の非修飾列名を使用して定義する必要があります。唯一、許可されたデータ・タイプは、double です。Cast 演算子を列式で使用することができますが、DB2 の暗黙的なキャストは可能ではありません。属性名は、文書モデルの属性名の規則に従う必要があります。索引のモデル定義ファイル内で定義される他の属性名とは異なる必要があります。

以下の規則を使用して、式の属性名を定義します。

- 列式の SQL AS 文節で明示的に指定した場合、指定した名前を使用する。例：ATTRIBUTES (C1+C2 AS myname)
- 指定された表の列が AS を指定せずに使用された場合、列の名前が使用される。例：CACHE TABLE (C1)
- 式が AS を指定せずに使用され、名前付き列を参照しない場合は、CREATE INDEX はエラーを報告します。

例：ATTRIBUTES (CAST(JULIAN_DAY(date) AS DOUBLE) as day, (price1+price2)/2 as avg_price)

引用符のない属性は英大文字にマップされ、検索時には大文字で指定する必要がありますので、注意してください。

CCSID ccsid

テキスト文書を索引付けする際に使用されるコード化文字セット ID。デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME='CCSID' です。

LANGUAGE language

リストについては、215 ページの『付録 E. サポートされている言語』を参照してください。デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME='LANGUAGE' です。

FORMAT format

列内の HTML などのテキスト文書のフォーマット。この情報は、テキスト文書の索引作成時に必要になります。構造化文書用にサポートされている文書フォーマットのリストについては、23 ページの『文書フォーマットおよびサポートされるコード・ページ』を参照してください。

構造化文書フォーマットの場合、文書モデル・ファイルで情報を指定できます。文書モデルが指定されない場合、文書のテキストは、デフォルトの文書モデルを使用して索引付けされます。172 ページの『文書モデル』を参照してください。

フォーマット・キーワードが指定されていない場合、デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME='FORMAT' です。

DOCUMENTMODEL documentmodel-name IN modelfilepath

modelfilepath は、モデル・ファイルのロケーションを指定します。

FORMAT 文節のフォーマット用のモデル定義も含まれます。これは、DB2 インスタンス所有者が読み取り可能なものでなければなりません。文書モデルを使用することにより、文書の特定のセクションを索引付けし、検索することができます。マークアップ・タグおよびセクション名も、文書モデルで定義できます。文書モデルは、HTML、XML、GPP 構造をサポートする文書フォーマットに結合されています。1 つのモデル・ファイルでは、1 つの文書モデルしか指定できません。

文書モデルを検索条件で参照する必要はないので、代わりにモデル・ファイルで指定されるすべてのセクション名を使用してください。文書モデルの詳細については、87 ページの『第 9 章 構造化文書を使用する作業』を参照してください。文書モデルは、CREATE INDEX コマンド実行時にのみ読み取られるので、それ以降のあらゆる変更は、この索引に関して認識されないことに注意してください。

分散 DB2 環境では、modelfilepath がすべてのノード上でアクセス可能になるようにファイル共有システムを使用してください。

USING CCSID ccsid

モデル・ファイルの内容を解釈する CCSID を指定します。デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME='MODELCCSID' です。

UPDATE FREQUENCY

索引更新の頻度は、更新が実行される時期を決定します。ユーザー表への変更が UPDATE MINIMUM オプションによって指定された値より小さい場合は、索引は更新されません。UPDATE FREQUENCY を指定しないと、デ

CREATE INDEX コマンド

フォルトの NONE が使用され、これ以上の索引の更新は行われません。これは、テキスト列にこれ以上の変更がない場合に有用です。

- **D** 索引が更新される曜日: * (毎日) または 0..6 (0=日曜日)
- **H** 索引が更新される時間: * (毎時) または 0..23
- **M** 索引が更新される分: 0..59
- **NONE** これ以上の索引更新はありません。更新は、手作業で開始する必要があります。

デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME='UPDATEFREQUENCY' です。

UPDATE MINIMUM minchanges

索引が UPDATE FREQUENCY によって自動的に更新される前に、テキスト文書に許可される最小の変更の数。正の整数値が許可されます。デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME='UPDATEMINIMUM' です。

この値は、DB2TEXT UPDATE コマンドでは、無視されることに注意してください。インクリメンタル更新の場合、ログ表およびトリガーがなければ変更の数を利用できないので、このオプションは、RECREATE INDEX ON UPDATE オプションと併用することはできません。

分散データベースの場合、UPDATE MINIMUM は各ノードでチェックされます。

REORGANIZE AUTOMATIC/MANUAL

REORGANIZE AUTOMATIC が指定されている場合、更新頻度を使用して実行される更新のみが、索引を認識します。このステップは、更新後、select REORGSUGGESTED from DB2EXT.TEXTINDEXES の値に従って自動的に完了します。

REORGANIZE MANUAL は、手動の UPDATE コマンド (REORGANIZE オプションを指定) と一緒になければ実行できません。

REORGANIZE 文節が省略された場合、デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME='AUTOMATICREORG' です。

REORGANIZE オプションの詳細に関しては、141 ページの『UPDATE INDEX』を参照してください。

REPLICATION CAPTURE TABLE capture-table-schema.capture-table-name CONTROL TABLE SCHEMA capture-control-schema

インクリメンタル更新処理の場合、通常は索引用に作成されるログ表の代わりに、指定されたレプリケーション・キャプチャー表が取られます。したがって、schemaname、tablename、およびレプリケーション・キャプチャー表名は、ローカル DB2 (フェデレーテッド) データベースのオブジェクトに関係しています。

capture-control-schema は、レプリケーション・コントロール表のスキーマ名です (例: ローカル DB2 上での IBMSNAP_PRUNE_SET)。レプリケーション・コントロール表は、レプリケーションを設定した後、ローカル DB2 システム上のニックネームとして使用可能である必要があります。

少なくとも、以下のキャプチャー・コントロール表で使用可能なニックネームがある必要があります。

- IBMSNAP_SIGNAL
- IBMSNAP_PRUNE_SET
- IBMSNAP_PRUNCNTL
- IBMSNAP_REGISTER
- IBMSNAP_REG_SYNC (DB2 以外のリモート・ソースのみ)

DB2 レプリケーション・センターはリモート・キャプチャー表とキャプチャー・コントロール表のローカル・ニックネームを必ずしも自動的に作成するわけではないため、この作業を手動にすることができます。この作業は、テキスト索引が作成される表のニックネームを作成することに似ています。

ユーザー表ニックネームとキャプチャー表の主キー列の列名は一致している必要があります。また、キャプチャー表ニックネームの列 IBMSNAP_OPERATION、IBMSNAP_COMMITSEQ、および IBMSNAP_INTENTSEQ の名前は変更しないようにしてください。

索引を作成した後、列名 DB2EXT.TEXTINDEXES(LOGVIEWNAME) と DB2EXT.TEXTINDEXES(LOGVIEWSCHEMA) は両方ともレプリケーション・キャプチャー表のローカル名を参照します。

Net Search Extender には DB2 レプリケーション・センターの機能のすべては必要ないので、変更データ表 (CD) または整合変更データ (CCD) 表は以下の規則に従っている必要があります。

- 変更キャプチャー登録を使用し、フル・リフレッシュ・コピー・オプションは使用しないでください。
- 変更のキャプチャーを水平サブセット化することは許可されていません。たとえば、トリガーを使用した方法はその一例です。「DB2 レプリケーションのガイドおよびリファレンス バージョン 8」の第 6 章『レプリケーション環境におけるデータのサブセット化』を参照してください。
- 列のサブセットの変更の登録が許可されるのは、主キー列、テキスト列、および DB2TEXT CREATE INDEX コマンドの属性およびキャッシュ表式に含まれるすべての列が含まれている場合だけです。
- キャプチャー表には主キー列が含まれていなければなりません。変更後イメージが十分であることに注意してください。
- キャプチャー表は圧縮できません。各主キーごとに、最新データのたった 1 つの項目が必要です。しかしながら、DB2 Net Search Extender では、全履歴が使用可能でなければなりません。
- 表は D/I オプションを使用する必要があります。これにより、ソース表上の主キーへの更新事項をトランスフォームして、挿入/削除の対にすることができます。

その他の前提条件には、以下のものがあります。

- 索引が作成されたソース表のサーバー・タイプおよびバージョンが、以下のいずれかであること。
 - DB2/AIX V8.1 以降
 - DB2/NT V8.1 以降

CREATE INDEX コマンド

- DB2/HP V8.1 以降
- DB2/LINUX V8.1 以降
- DB2/SUN V8.1 以降
- DB2 z/OS[®] V7.2 以降
- DB2 OS/400[®] V5.2 以降
- Informix[®] IDS 9.3
- ORACLE 9i
- SYBASE ASE 12.5
- Microsoft SQL Server 2000
- サポートされるラッパーには、以下のものが含まれています。
 - DB2: DRDA[®]
 - Informix: Informix
 - ORACLE: NET8, (SQLNET)
 - SYBASE: CTLIB
 - MSSQLSERVER: MSSQLODBC3

注意および制約事項

正しいソース表名を登録表に挿入するようにしてください。リモート DBMS の種類によって、リモート表名を使用しなければならない場合と、ローカル・ニックネームを使用しなければならない場合があります。

- DB2: リモート表名 (リモート・サーバー上の表名)
- DB2 以外: ローカル・ニックネーム (フェデレーテッド DB2 データベース内の対応するニックネーム)

ローカル・ユーザーがニックネームを使用してリモート・データ・ソースにアクセスできるようにユーザー・マッピングがなければならず、リモート・ユーザーは表に対してコントロール特権を持っている必要があります。

DB2 インスタンス所有者ユーザー ID がローカル・ユーザー ID と異なる場合、DB2 インスタンス所有者ユーザー ID 用の追加ユーザー・マッピングが必要です。

指定された基本表名は、1 つのニックネームに対するビューであってはなりません。それは、ビューは複数のニックネームをカバーすることができ、複数の CD および CCD 表を含めることもできるからです。レプリケーション・キャプチャー文節には CD または CCD 表を 1 つだけ指定することもできるので、ニックネームのビューはサポートされていません。また、リモート・ビューのニックネームは、主キーが欠落しているため、サポートできません。

CD または CCD 表はニックネームでなければならず、ビューまたは別名であってはなりません。

「DB2 レプリケーションのガイドおよびリファレンス バージョン 8」については、viii ページの『関連情報』を参照してください。

COMMITCOUNT FOR UPDATE count

インクリメンタル更新処理の場合、コミット・カウントを指定できます。詳細に関しては 141 ページの『UPDATE INDEX』を参照してください。指定されない場合は、デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME='COMMITCOUNT' です。

索引の COMMITCOUNT FOR UPDATE 値は、DB2EXT.TEXTINDEXES.COMMITCOUNT にあります。これは、ALTER INDEX コマンドを使用して、索引ごとに変更できます。また、UPDATE FREQUENCY 指定に従ってスケジュール済みの更新処理にも適用されます。0 の値は、更新が 1 つのトランザクションで完了したことを意味し、0 より大きい値は、1 つのトランザクションで処理する文書の数を指定します。

commitcount を使用すると、パフォーマンスに影響します。詳しくは、45 ページの『パフォーマンスに関する考慮事項』を参照してください。

RECREATE INDEX ON UPDATE

これは、インクリメンタル索引更新を許可せず、更新操作が実行される際に索引を再作成します (コマンドまたはスケジュールされた更新により)。詳細に関しては 141 ページの『UPDATE INDEX』の「使用上の注意」を参照してください。

注

ユーザー表ではトリガーは作成されず、ログ表も作成されません。

INDEX DIRECTORY directory

テキスト索引の保管先ディレクトリー・パス。ディレクトリーは索引データを含むので、ディレクトリーには、DB2 インスタンス所有者ユーザー ID 用の読み取り/書き込み、および実行権限があることを確認してください。

デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME=INDEXDIRECTORY' です。サブディレクトリー NODE<nr> は、サーバーの論理ノード上の索引を区別するために、このディレクトリーの下に作成されます。

分散 DB2 環境では、このディレクトリーはすべての物理ノード上に存在する必要があることに注意してください。

WORK DIRECTORY directory

別個の作業ディレクトリーをオプションで指定し、索引検索および管理操作時に一時ファイルの保管に使用することができます。ディレクトリーが存在し、DB2 インスタンス所有者ユーザー ID に対して読み取り/書き込みおよび実行権限を持っている必要があります。

デフォルト値は、DB2EXT.DBDEFAULTS ビューから取られます。ここで、DEFAULTNAME='WORKDIRECTORY' です。サブディレクトリー NODE<nr> は、サーバーの論理ノード上の索引を区別するために、このディレクトリーの下に作成されます。

分散 DB2 環境では、このディレクトリーはすべての物理ノード上に存在する必要があることに注意してください。

ADMINISTRATION TABLES IN tablespace-name

索引用に作成された管理表のための REGULAR 表スペースの名前。表スペースが存在する必要があります。指定されない場合は、索引が基本表上に作成されていれば、ユーザー表の表スペースが選択されます。

ニックネームまたはビューの場合は、デフォルトの表スペースが DB2 によって選択されます。

分散 DB2 環境でビューやニックネームのテキスト索引、またはストアード・プロシージャ検索のためのテキスト索引を作成する場合、表スペースは単一ノードでなければなりません。

CACHE TABLE (SQL-column-expression-list)

指定された列式からなるキャッシュ表が索引に追加して作成されます。このキャッシュは、フルテキスト検索の結果を DB2 表に結合しないで、ストアード・プロシージャ検索を介して戻すために使用されます。CONTAINS 関数をもったフルテキスト索引を使用する通常の DB2 検索は、常に可能です。

索引が作成される表の非修飾列名を使用して SQL 列式を定義します。許可される SQL 列式タイプは、すべて組み込み、およびユーザー定義特殊タイプです。結果セットの列名は、次の規則を使用して決定されます。

- 列式の SQL AS 文節で明示的に指定された場合、指定された名前が使用される。例：CACHE TABLE (C1+C2 AS myname)
- 指定された表の列が AS 文節を指定せずに使用された場合、列の名前が使用される。例：CACHE TABLE(C1)
- 式が AS を指定せずに使用され、名前付き列を参照しない場合は、CREATE INDEX はエラーを報告します。
- 列名の重複は許可されていません。

CLOB データ・タイプはキャッシュ・データ・タイプとしてサポートされていません。CLOB データ・タイプは VARCHARS にキャストする必要があります。

注

結果セットの列名を分けて指定できない場合、CREATE INDEX コマンドはエラーを戻します。また、キャッシュ表は、作成後、暗黙的にはアクティブ化されません。たとえばストアード・プロシージャによる検索は、DB2TEXT ACTIVATE CACHE が実行されるまでは、不可能です。

このオプションは、ユーザー表が単一ノード表スペースに保管された場合のみ、分散 DB2 環境によって使用されます。

PERSISTENT IN directory

キャッシュも永続的に作成され、非アクティブ化またはシステム・リブートの直後にアクティブ化できるように指定します。永続キャッシュは、指定されたディレクトリーに保管されます。

ディレクトリーが指定されていない場合、デフォルト値は `db2ext.dbdefaults` ビューから取られます。ここで、`DEFAULTNAME='CACHEDIRECTORY'` です。

TEMPORARY

キャッシュが永続的に保管されないように指定します。PERSISTENT または TEMPORARY が指定されない場合、デフォルト値は、`DB2EXT.DBDEFAULTS` ビューから取られます。ここで、`DEFAULTNAME='USEPERSISTENTCACHE'` です。

MAXIMUM CACHE SIZE memsize

DB2TEXT ACTIVATE CACHE 時に作成されるキャッシュ表の最大サイズを指定します。memsize パラメーターは、メガバイト単位で正の整数として指定する必要があります。memsize にはデフォルト値がありません。整数が小さ過ぎると、ACTIVATE CACHE コマンドは失敗します。実際のキャッシュ・サイズは ACTIVATE CACHE コマンド時に計算されます。

プラットフォームごとの最大キャッシュ・サイズの限度は、以下のとおりです。

- Windows: 1024 MB (1 GB = 1073741824 バイト)
- AIX: 1536 MB (1.5 GB = 1610612736 バイト)
- Solaris、Linux、HP-UX: 2048 MB (2 GB = 2147483647 バイト)

詳細については、195 ページの『付録 B. 大量のメモリーの使用』を参照してください。

PCTFREE percentage

指定した割合のキャッシュを文書の追加のために、フリーのまま保持されるように指定します。割合は 0 以上 100 未満の整数値でなければなりません。指定されない場合、デフォルト値は `db2ext.dbdefaults` ビューから取られます。ここで、`DEFAULTNAME='PCTFREE'` です。

詳細については 112 ページの『ACTIVATE CACHE』を参照してください。

INITIAL SEARCH RESULT ORDER (SQL-order-by-list)

初期の索引付け時にユーザー表の内容を検索するために使用される順序を指定します。このオプションを使用して、フル・テキスト検索結果の動的ランキングをスキップすると、キャッシュ結果表に保管されているとおりの、索引付けされた順序で文書が戻されます。

詳細については 167 ページの『第 16 章 ストアード・プロシージャ検索関数』を参照してください。

事前ソートされた索引を使用して作業できるのは、ストアード・プロシージャ検索インターフェースを使用している場合だけです。

注

インクリメンタル更新後は、新規または変更された文書に対して、索引順序を保証できません。たとえば: INITIAL RESULT
ORDER(length(column1) asc, column2+column3 desc)

KEY COLUMNS FOR INDEX ON VIEW (SQL-columnname-list)

ビュー上に索引が作成された場合は、KEY COLUMNS FOR INDEX ON VIEW 文節を指定する必要があります。作成されない場合は、指定「しないで」ください。列名のリストは、ビュー内の行を「一意に」識別する列を指定します。

この一意性は、主キーの場合とは異なって DB2 によるチェックができないので、ユーザーが同等の一意性を保証する責任を持っています。指定された列は、索引用のログ表の一部を作成します。

INDEX CONFIGURATION (option-value), ...

索引構成値は次のとおりです。デフォルト値は、下線が引かれています。

オプション	値	説明
TreatNumbersAsWords	<u>0</u> または 1	数字のシーケンスが文字に隣接していても、別個のワードとして解釈する。たとえば、デフォルトの 0 は、tea42at5 が 1 つのワードと見なされることを意味する。
IndexStopWords	<u>0</u> または 1	索引付け中にストップワードを考慮するか、無視する。現在、ストップワード・リストは、ディレクトリー <instance>/sql1lib/db2ext/resources の UCS-2 ファイル <language>.tsw にある。このファイルを変更しても、索引作成後は影響はない。また、<language> は、CREATE INDEX コマンドからの LANGUAGE 値であることに注意。
UpdateDelay	秒数	キャプチャー表なしのインクリメンタル更新の所要時間を秒単位で指定する。この所要時間を超過した古い項目だけが、ログ表から取り去られる。これは更新が失われることのないようにするためのものである。たとえば、ユーザー・トランザクションが更新コマンドに干渉するというトランザクションのシナリオで、文書の変更が反映されずに終わることがないようにするためのものである。したがって、UpdateDelay パラメーターは、索引を作成した表に対して実行される、ユーザー書き込みトランザクションの最大所要時間に設定する必要がある。

CONNECT TO database-name

このコマンドのターゲットであるデータベースの名前。DB2DBDFT が設定されており、ユーザーがサーバーでコマンドを実行している場合は、このパラメーターを省略できます。ユーザー ID には必須 DB2 許可がなければならないことに注意してください。

USER userid USING password

password および userid を使用してデータベースに接続します。指定しない場合は、パスワードなしで現行ユーザー ID から接続が試行されます。

データベースへの変更

- DB2 Net Search Extender カタログ・ビューを変更します。

- 指定された表スペースに索引ログ表を作成します。これは、RECREATE INDEX オプションが指定されておらず、キャプチャー表も指定されていません。
- 指定された表スペースに索引イベント表を作成します。
- 最初の更新の実行時まで据え置き: ユーザー・テキスト表上にトリガーを作成します (RECREATE INDEX が指定されておらず、キャプチャー表も使用されていない場合のみ)。
- レプリケーション・キャプチャー表が使用されている場合、次の変更がキャプチャー・コントロール表に加えられます。
 - IBMSNAP_PRUNCTNL および IBMSNAP_PRUNE_SET 表への挿入
 これらの表の項目は、以下の列によって固有に識別されます。
 - APPLY_QUAL='NSE' || <DB2 instance running NSE>
 - SET_NAME= <internal index identifier>
 - TARGET_SERVER=<DB2 database name target to DB2TEXT operation>
 列の詳細については、34 ページを参照してください。

共有メモリーへの変更

ACTIVATE の実行時まで据え置き: CACHE TABLE 文節が使用された場合、結果表のキャッシュは、共有メモリー に作成されます。

ファイル・システムへの変更

- サブディレクトリー NODE<nr> が、索引、作業、およびキャッシュ・ディレクトリーの下に作成されます。
- ディレクトリー <internal index name> が <indexdirectory>/NODE<nr> の下に作成されます。この場合、indexdirectory は、このコマンドの対応するパラメーターを参照し、NODE<nr> は、分散 DB2 環境のノード番号に関係付けられます。

使用法

フル・テキスト索引の作成には、ユーザー表上の主キーが必要です。DB2 Net Search Extender バージョン 8.1 では、マルチコラム DB2 主キーは、タイプの制限なく使用できます。しかし、表値検索を使用する場合、コンパウンド主キーは許可されません。

主キー列の数は、14 に制限され、すべての主キー列の全長は、 $1024 - 14 = 1010$ バイトに制限されています。

- ATTRIBUTES、CACHE TABLE および INITIAL SEARCH RESULT ORDER の SQL 式の合計サイズは、24K バイトを超えてはなりません。
- 初期の索引更新は、常に 1 つの論理トランザクションとして実行されるので、この場合のコミット・カウントはありません。

CREATE INDEX コマンド

注

索引の作成後は、主キー列またはビュー・キー列の長さを ALTER TABLE コマンドによって変更してはなりません。

ユーザー表、フル・テキスト索引およびキャッシュ結果表の間の同期は、索引更新コマンド中に完了します。詳細については 141 ページの『UPDATE INDEX』を参照してください。

DEACTIVATE CACHE

このコマンドは、キャッシュ表を解放します。永続キャッシュは、次の ACTIVATE コマンドで再利用するために保持されます。次のアクティブ化まで、ストアード・プロシージャを介した検索操作は、非活動キャッシュでは不可能です。

許可

DB2 カタログ・ビューによると、このコマンドのユーザー ID は、フル・テキスト索引が作成された表に対して CONTROL 特権が必要です。

コマンド構文

```
DEACTIVATE-CACHE-FOR-INDEX index-schema-"." index-name FOR-TEXT
|connection-options|
```

connection-options:

```
CONNECT-TO database-name USER userid USING password
```

コマンド・パラメーター

index-schema

CREATE INDEX コマンドで指定されたテキスト索引のスキーマ。スキーマが指定されていない場合は、DB2 接続のユーザー ID がスキーマ名として使用されます。

index-name

CREATE INDEX コマンドで指定されたテキスト索引の名前。

CONNECT TO database-name

このコマンドのターゲットであるデータベースの名前。DB2DBDFT が設定されており、ユーザーがサーバーでコマンドを実行している場合は、このパラメーターを省略できます。ユーザー ID には必須 DB2 許可がなければならぬことに注意してください。

USER userid USING password

password および userid を使用してデータベースに接続します。指定しない場合は、パスワードなしで現行ユーザー ID から接続が試行されます。

使用法

次のコマンドのいずれかが索引上で実行中の場合には、このコマンドを発行できないことに注意してください。

- ACTIVATE CACHE
- DEACTIVATE CACHE
- UPDATE INDEX
- ALTER INDEX

DEACTIVATE CACHE コマンド

- DROP INDEX
- CLEAR EVENTS

注

永続キャッシュの非アクティブ化後、ストアード・プロシージャーによる検索に対して、キャッシュはアクセス不能になります。ただし、その間に更新が実行されない限り、高速に **ACTIVATE** することが可能です。

非アクティブ化中に更新が行われた場合、永続キャッシュは、**ACTIVATE CACHE** コマンドを使用して自動的に最初から再作成されます。

DROP INDEX

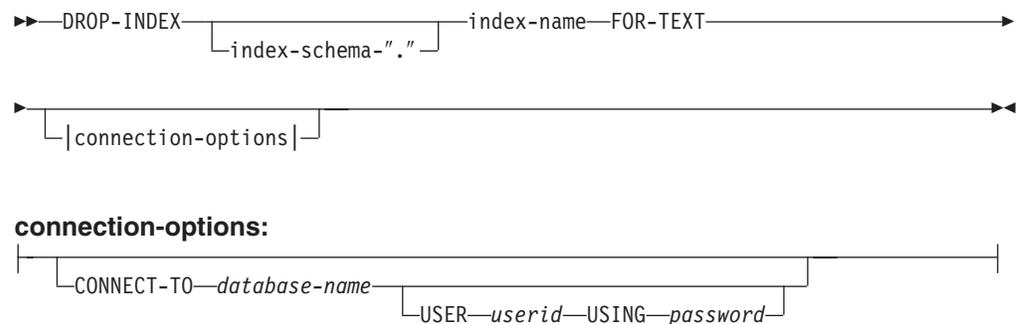
このコマンドは、テキスト列用のフル・テキスト索引をドロップします。索引のキャッシュがアクティブ化されている場合は、このコマンドを使用して削除されません。

許可

DB2 カタログ・ビューによると、このコマンドのユーザー ID は、フル・テキスト索引が作成された表に対して CONTROL 特権が必要です。あるいは、ユーザーはデータベース管理者 (DBADM) となることができます。

あるいは、データベース管理者 (DBADM) は、FORCE オプションを使用してデータベースを使用不可にできるので、索引をドロップできます。

コマンド構文



コマンド・パラメーター

index schema

CREATE INDEX コマンドで指定されたテキスト索引のスキーマ。スキーマが指定されていない場合は、DB2 接続のユーザー ID がスキーマ名として使用されます。

index-name

CREATE INDEX コマンドで指定された索引の名前。索引スキーマとともにデータベースのフル・テキスト索引を一意に識別します。

CONNECT TO database-name

このコマンドのターゲットであるデータベースの名前。DB2DBDFT が設定されており、ユーザーがサーバーでコマンドを実行している場合は、このパラメーターを省略できます。ユーザー ID には必須 DB2 許可がなければならぬことに注意してください。

USER userid USING password

password および userid を使用してデータベースに接続します。指定しない場合は、パスワードなしで現行ユーザー ID から接続が試行されます。

使用法

キャッシュ表のアクティブ化状況に無関係に索引は削除されます。詳細については、112 ページの『ACTIVATE CACHE』を参照してください。

DROP INDEX コマンド

次のコマンドのいずれかが索引上で実行中の場合には、コマンドを発行できないことに注意してください。

- UPDATE INDEX
- CLEAR EVENTS
- ALTER INDEX
- ACTIVATE CACHE
- DEACTIVATE CACHE
- DROP INDEX

注

DB2 のユーザー表がドロップされる前か後に、索引を手動でドロップする必要があります。ドロップしない場合は、結果が正しくクリーンアップされません。

データベースへの変更

- DB2 Net Search Extender カタログ・ビューを変更します。
- DB2 索引をドロップします。
- 索引ログ/イベント表をドロップします。
- ユーザー・テキスト表上のトリガーを削除します。

レプリケーション・キャプチャー表を使用する場合、
IBMSNAP_PRUNE_SET および IBMSNAP_PRUNCTRNL 表内の項目が除去されます。

共有メモリーへの変更

キャッシュ表が削除されます。

ファイル・システムへの変更

- ディレクトリー <internal index name> が、索引およびドロップされた索引の作業ディレクトリーから削除されます。
- 索引の永続キャッシュを削除します。

DB2EXTTH (ユーティリティー)

この独立ユーティリティーは、シソーラス定義ファイルをコンパイルします。シソーラス・コンパイラーを実行した後で、検索引き数構文のシソーラス関連機能を使用できます。

許可

なし。このコマンドは、必ずしも表所有者を制限しませんが、照会の状況においてのみ意味をなします。

コマンド構文

```

▶▶ db2extth -ccsid—code page— -f—definition-file-name ▶▶
├──quiet──┤
├──-h──┤
├──-H──┤
├──-?──┤
└──-copyright──┘

```

コマンド・パラメーター

-f definition-file-name

シソーラス定義を含むファイルの名前。ファイル名には、絶対パスまたはファイルへの相対パスが含まれる必要があります。ファイル名は、8+3 文字に制限されており、拡張子はオプションです。

シソーラス・ディクショナリーは、定義ファイルと同じディレクトリーに作成され、同じ名前を持ちます。唯一の違いは、ディクショナリーには、次の拡張子がある点です。wdf、wdv、grf、grv、MEY、ROS、NEY、SOS、および Ikn。ただし、n は数字。既存のシソーラス・ファイルが同じ名前を持っていた場合、上書きされることに注意してください。

-ccsid code page

シソーラス定義ファイルが書き込まれるコード・ページ。シソーラス用にサポートされているコード・ページに関しては、261 ページの『付録 L. シソーラスでサポートされている CCSID』を参照してください。

-quiet 出力情報は、表示されません。

-copyright

製品の内部ビルド番号を戻します。問題を報告する場合は、この番号を使用します。

-h、-H、または -?

ヘルプ情報を表示します。

使用法

このコマンドを使用して、シソーラス定義ファイルをバイナリー・シソーラス定義フォーマットにコンパイルします。

注

フォーマットは、DB2 テキスト情報エクステンダー バージョン 7.2 と同じですが、以下の点に変更されました。

- 新規リレーションシップ **BROADER** および **NARROWER** は、従来、テキスト情報エクステンダーで使用されていた **HIGHER_THAN** および **LOWER_THAN** のリレーションシップと同等です。これらの古い検索リレーションシップを参照するためには、新規のリレーションシップを使用する必要があります。
- シソーラスが照会において完全修飾されていない限り、検索中に使用可能にするためには、シソーラス・ディクショナリー・ファイルを `<os-dependent>/sql1lib/db2ext/thes` に保管する必要があることにも注意してください。

詳細に関しては、89 ページの『第 10 章 検索項目を拡張するためのシソーラスの使用』を参照してください。

UPDATE INDEX

このコマンドは、即時に索引付け処理を開始し、索引に関連付けられたテキスト列の現在の内容を反映し、索引を最新の状態に更新します。

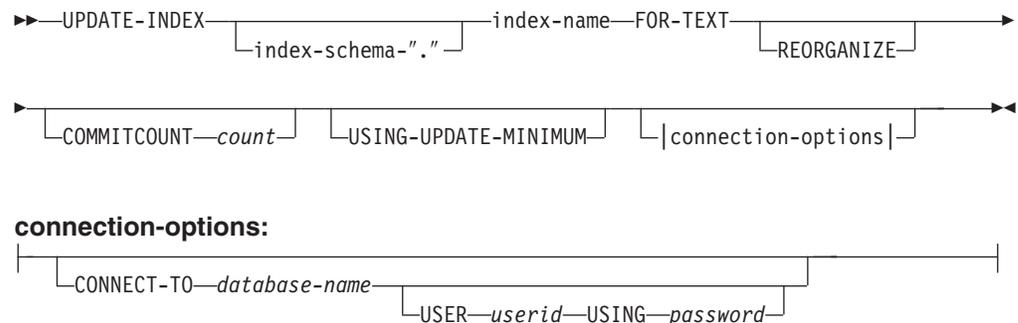
更新の実行中も、CONTAINS 述部を使用した検索が可能です。アクティブ化されたキャッシュ結果表を持つ索引の場合、更新時にストアード・プロシージャによる検索も可能です。ただし、変更されたテキストがフル・テキスト索引にまだコミットされていないにもかかわらず、キャッシュ表の列が新しい値を表示する可能性があります。

CREATE INDEX コマンドで RECREATE INDEX ON UPDATE オプションを使用すると、再作成前に索引がクリアされます。更新が完了するまで、空の結果が戻されます。

許可

DB2 カタログ・ビューによると、このコマンドのユーザー ID は、フル・テキスト索引が作成された表に対して CONTROL 特権が必要です。

コマンド構文



コマンド・パラメーター

index-schema

テキスト索引のスキーマ。これは CREATE INDEX コマンドで指定されます。スキーマが指定されていない場合は、DB2 接続のユーザー ID が使用されます。

index-name

テキスト索引の名前。これは CREATE INDEX コマンドで指定されます。

REORGANIZE

テキスト列が頻繁に更新されると、索引へのその後の更新が非効率になる可能性があります。更新処理を再び効率的にするには、索引を再編成します。索引が再編成を必要としているかどうかを判別するには、DB2EXT.TEXTINDEXES ビューを使用します。

手動による索引のチェックおよび再編成を回避するには、CREATE INDEX コマンドの REORGANIZE AUTOMATIC オプションを使用します。

注

再編成処理は、通常の更新の後に行われます。

USING UPDATE MINIMUM

指定した変更の数に達した場合に限り、CREATE INDEX コマンドから UPDATE MINIMUM 設定値を使用して、インクリメンタル更新を開始します。デフォルトでは、無条件に更新を開始します。

分散データベースの場合、UPDATE MINIMUM は各ノードでチェックされます。

詳細については、120 ページの『CREATE INDEX』を参照してください。

COMMITCOUNT count

INTEGER 値 ≥ 0 は、インクリメンタル索引更新のため、検索エンジンおよび DB2 により 1 つのトランザクションで処理される文書の数を表示します。

ただし、CREATE INDEX コマンドの後の最初の更新、または RECREATE INDEX ON UPDATE オプションによる更新などの初期更新の場合、COMMITCOUNT を無視する論理トランザクションが 1 つだけ存在します。これは、ALTER INDEX コマンドを使用して変更できます。

CONNECT TO database-name

このコマンドのターゲットであるデータベースの名前。DB2DBDFT が設定されており、ユーザーがサーバーでコマンドを実行している場合は、このパラメーターを省略できます。ユーザー ID には必須 DB2 許可がなければならぬことに注意してください。

USER userid USING password

password および userid を使用してデータベースに接続します。指定しない場合は、パスワードなしで現行ユーザー ID から接続が試行されます。

使用法

このコマンドは同期的に実行します。分散 DB2 環境内のすべての必要な DB2 論理/物理ノード上で更新処理を開始します。所要時間は、索引付けするファイル数およびすでに索引付けされた文書数によって異なります。更新の状況は、索引ごとに作成されるビューを介して表示することができます。このビューの名前は、列 EVENTVIEWNAME の DB2EXT.TEXTINDEXES から検索できます。詳細に関しては 199 ページの『付録 C. Net Search Extender インフォメーション・カタログ』を参照してください。

処理済みのコミット文書の数を表示するには、2 つのオプションがあります。更新が引き続き実行中であるか、どのくらいの文書が索引にコミット済みであるかを判別するには、DB2EXT.TEXTINDEXES (NUMBER_DOCS) ビューを使用します。変更の開始、コミット、および更新処理の完了に関する情報も、索引に関連したイベント・ビューを使用します。

処理することになっている未コミット文書の数を表示するには、CONTROL LIST ALL LOCKS FOR INDEX コマンドを使用します。

注

接続したノードから情報を表示できるのは、これらのビューだけです。

物理ノードで基本表にインクリメンタル更新を行う場合、各ノードの時刻が同期していなければなりません。時刻が同期していないと、更新が失われたり、まったく実行されなかったりする恐れがあります。

次のコマンドのいずれかが索引上で実行中の場合には、このコマンドを発行できないことに注意してください。

- CLEAR EVENTS
- ALTER INDEX
- DROP INDEX
- ACTIVATE CACHE
- DEACTIVATE CACHE
- UPDATE INDEX

非アクティブ化された永続キャッシュ結果表を使用して索引を更新した後、永続キャッシュは削除され、次の ACTIVATE CACHE コマンドによって、データベースの内容に基づいて再作成されます。

ユーザーがこのコマンドに割り込んだ場合、更新機能に関する処理はすべて停止します。コミット・カウントがインクリメンタル更新で使用された場合は、一部の更新は索引内で見ることができるものの、他の更新は新たな更新コマンドが必要な場合があります。

索引の自動更新を停止するには、更新サービスに使用されるパーティション上で索引更新コマンドを実行中の DB2 インスタンス所有者処理を検索します。すべてのパーティション上でこの処理および更新処理を停止します。

注

コマンドは、すべてのパーティションでの索引作成および初期の索引更新のために 2 つの別個のフェーズで機能するため、db2text drop index コマンドを発行して、索引が一部しか利用できないことがないようにします。このコマンドが発行されない場合は、次の更新 (更新コマンドまたは更新頻度オプションによって起動される可能性がある) は、一貫性のある状態を確保するために完全な再索引付けを実行します。

データベースへの変更

- イベント表への挿入
- 索引ログ表からの削除

レプリケーション・キャプチャー表を使用する場合、データベースに以下の変更が加えられます。

UPDATE INDEX コマンド

- 最初の更新が開始する前に、IBMSNAP_SIGNAL 表にシグナルが追加される。
- インクリメンタル更新後に、IBMSNAP_PRUNE_SET の同期点が変更される。

HELP

このコマンドは、使用可能な DB2TEXT コマンドのリスト、または個別の DB2TEXT コマンドの構文を表示します。

許可

不要。

コマンド構文



コマンド・パラメーター

HELP または ?

指定したコマンドまたは理由コードのヘルプを表示します。

command

DB2TEXT コマンドを示す最初のキーワードです。以下のとおりです。

- ENABLE
- DISABLE
- CREATE
- DROP
- ALTER
- UPDATE
- CLEAR
- START
- STOP
- CONTROL
- ACTIVATE
- DEACTIVATE

reasoncode

DB2 Net Search Extender コマンドからの理由コード。

使用法

最初のキーワード以外にもキーワードが指定された場合、残りのキーワードは無視され、識別されたコマンドの構文が表示されます。

'?' または 'HELP' の後に 'command' パラメーターが指定されない (または、パラメーターがまったく指定されない) 場合は、DB2TEXT は選択可能な DB2TEXT コマンド・パラメーターをすべてリストします。

COPYRIGHT

Net Search Extender の製品および著作権の情報を表示します。

許可

不要。

コマンド構文



コマンド・パラメーター

COPYRIGHT / LEVEL

製品のバージョンの著作権文、バージョン番号およびビルド情報を表示します。

第 14 章 検索引き数の構文

検索引き数は、テキスト文書で用語を検索する際に指定する条件です。検索パラメーターと 1 つ以上の検索語から構成されています。

検索引き数の例は、75 ページの『SQL 検索引き数の指定』 および search という名のファイルに記載されています。80 ページの『その他の検索構文の例』を参照してください。

検索引き数を使用する SQL スカラー検索関数は、次のとおりです。

CONTAINS

この関数は、特定のテキスト文書でテキストを検索するために検索引き数を使用します。文書にテキストまたは検索引き数に指定された関係が含まれている場合は、INTEGER 値 1 を戻します。それ以外の場合は、0 を戻します。

NUMBEROFMATCHES

この関数は、テキスト文書内で検索するために検索引き数を使用し、文書ごとに一致した結果の数を示す INTEGER 値を戻します。

SCORE

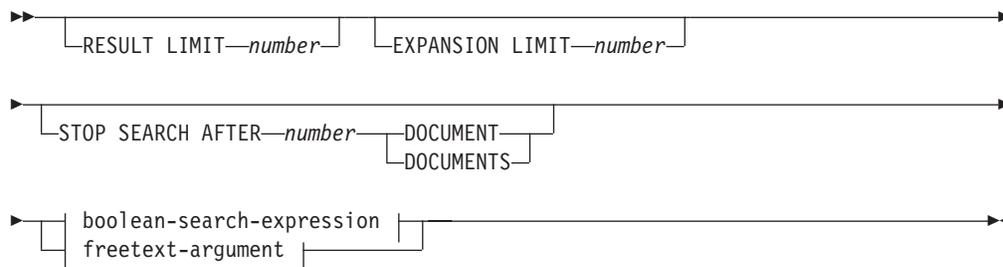
この関数は、テキスト文書内で検索するために検索引き数を使用します。検出された文書ごとに値を戻し、検出された文書が検索引き数によってどの程度適格に記述されているかを示します。

注

ストアード・プロシージャ検索および SQL 表値関数の検索引き数には同じ構文を使用します。

検索引き数

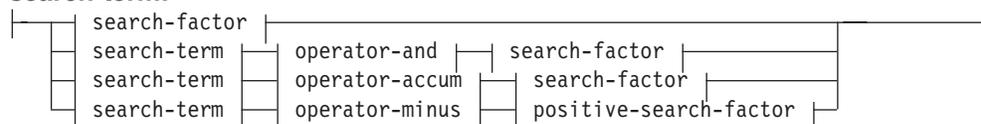
検索引き数の構文



Boolean-search-expression:



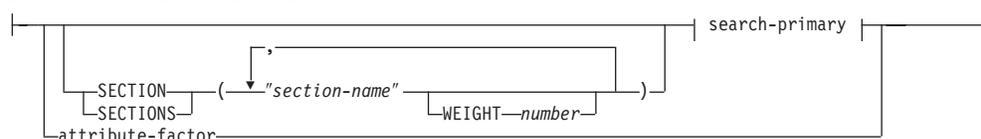
search-term:



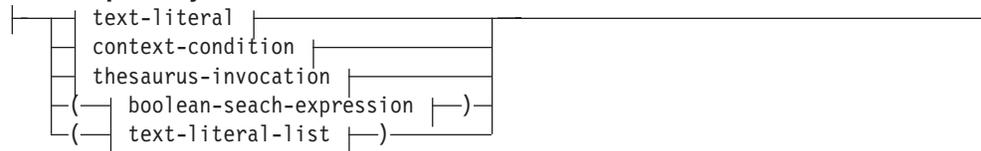
Search-factor:



Positive-search-factor:



Search-primary:



Operator-and:



Operator-or:



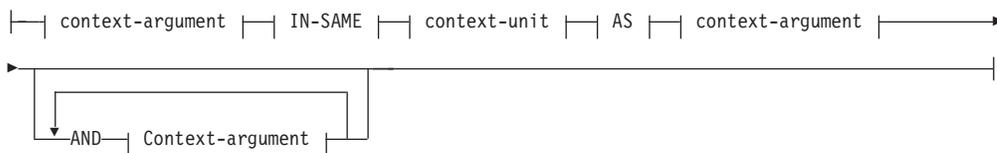
Operator-accum:



Operator-minus:



Context-condition:



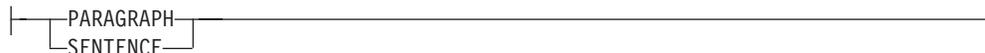
Context-argument:



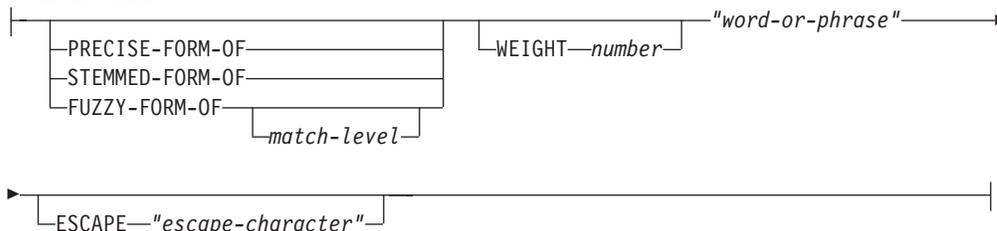
Text-literal-list:



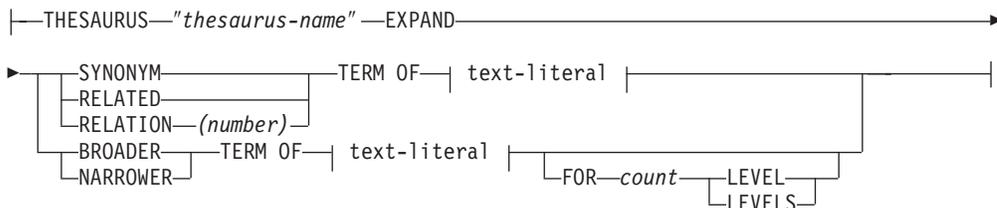
Context-unit:



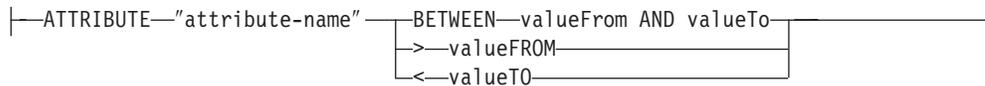
Text-literal:



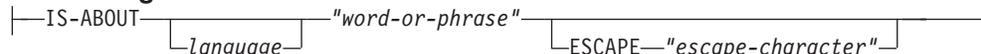
thesaurus-invocation:



Attribute-factor:



freetext-argument:



例

例は、75 ページの『SQL 検索引き数の指定』に記載されています。

検索パラメーター

RESULT LIMIT number

フル・テキスト検索によって戻される結果の最大数を指定するキーワード。
 RESULT LIMIT は SCORE 関数とともに使用して、戻される結果にスコアが付けられ、最もよい結果だけが処理されるようにします。

EXPANSION LIMIT number

検索のために語を拡張する回数の最大数を指定するキーワード。たとえば、検索語 'a*' を何回拡張するかを決める場合に使用します。

STOP SEARCH AFTER number DOCUMENTS(S)

検索しきい値を指定するキーワード。検索中に文書の出現数がこの値に達すると、検索は停止し、中間結果が戻されます。値が低い場合は、検索のパフォーマンスが上がるものの、結果が少なくなり、潜在的にランクの高い文書が省略される可能性があります。

デフォルト値はなく、*number* 値は、正の整数でなければなりません。

boolean-search-expression

検索用語および検索要因は、構文図に従って、ブール演算子 NOT、AND、OR、ACCUM および MINUS を使用して、結合することができます。演算子には、次の優先順位があります (優先順位の高い順に): NOT > MINUS = ACCUM = AND > OR。これについて次の例を参照してください。

"Pilot" MINUS "passenger" & "vehicle" | "transport" & "public"

は、以下のように評価されます。

((("Pilot" MINUS "passenger") & ("vehicle"))) | ("transport" & "public")

ブール引き数の 1 つが真であると評価した場合、演算子 ACCUM は真であると評価します (これは、OR 演算子に類似しています)。ランク値は、両方のオペランドからランク値を累算して計算されます。ACCUM 演算子は、AND と同じバイnding (優先順位) を持っています。演算子 MINUS は、左のオペランドが真であると評価されると、真であると評価されます。ランク値は、右のオペランドが真であると評価されると、左のオペランドのランク値を取り、ペナルティーを減算することによって計算されます。

search-primary

thesaurus-invocation からなる search-primary は、(文書の指定されたセクションに) 拡張された text-literal が検出された場合に真と評価されます。
 text-literal-list からなる search-primary は、(文書の指定されたセクションに) text-literals が検出された場合に真と評価されます。

SECTION(S) section-name

検索がその内部に制限される構造化文書で、1 つまたは複数のセクションを指定するキーワード。セクション名は、索引作成時に指定されたモデル・ファイルに指定する必要があります。120 ページの『CREATE INDEX』を参照してください。

セクション名には、大文字小文字の区別があります。モデル・ファイルと照会でのセクション名が同じになるようにしてください。

このモデルは、識別可能なセクションを含む文書の構造を記述しているので、これらのセクションの内容を個別に検索することができます。セクション名は、マスク文字を使用してマスクすることはできません。セクションの 1 つで `search-primary` が検出された場合、SECTION 文節を使用する `positive-search-factor` は真であると評価されます。

context-argument IN SAME context-unit AS context-argument AND context-argument ...

この条件を使用すれば、`text-literal` の組み合わせが同じ段落または文に出現する箇所を検索できます。コンテキスト引き数は、つねに `text-literal-list` に同等であり、シソーラス拡張を使用して `text-literal` をそのようなりストに拡張することができます。

それぞれの拡張された `context-argument` の少なくとも 1 つの `text-literal` を含む `context-unit` (段落、次いで、文) が文書にある場合、条件は真であると評価されます。これについて次の例を参照してください。

```
("a","b") IN SAME PARAGRAPH AS ("c","d")
    AND THESAURUS "t1" EXPAND SYNONYM TERM OF "e".
```

`e1`、`e2` が `e` の同義語であると想定すると、次の段落が一致します。

```
".. 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

この **PRECISE FORM OF** の後に指定する語 (または熟語中のそれぞれの語) を、入力したとおりの形で検索するよう指定するキーワード。この形式の検索は、大文字小文字を区別します。つまり、大文字および小文字の使用が有効です。たとえば、`mouse` を検索する場合、“`Mouse`” は検出されません。

STEMMED FORM OF

この **STEMMED FORM OF** の後に指定する語 (または熟語中のそれぞれの語) を、まずその語幹に縮めてから検索を実行するよう指定するキーワード。この検索形式では大文字小文字が区別されません。たとえば、`mouse` を検索する場合、“`Mouse`” も検出されます。

語が語幹形式に縮められる方法は、言語によって異なります。現在、英語のみがサポートされており、語は、規則的な語尾変化に従う必要があります。

FUZZY FORM OF

「ファジー」検索のためのキーワードです。検索項目とつづりの類似した語を検索します。これは、光学式文字認識 (OCR) プログラムで作成した文書を検索する場合に特に役に立ちます。そのような文書には、つづりの誤った語が含まれていることがあります。たとえば、OCR プログラムは、`economy`

という用語を `economy` と認識することがあるかもしれません。最初の 3 文字が一致する必要があること、および検索要素内の語にマスク文字が含まれている場合、ファジー検索は使用できないことに注意してください。

match level

類似性の度合いを指定する 1 ~ 100 の整数。100 の方が 1 よりも類似性が高いことを表します。100 は「完全一致」を指定し、60 は、非常に「ファジーな値」と見なされます。突き合わせレベル (`match level`) がファジーなほど、より多くの文書が検索に対して適合するので、検索に要する時間が長くなります。デフォルトの `match level` は 70 です。

WEIGHT number

`text-literal` を重み値に関連付けてデフォルトのスコアを変更します。許可された重み値は、0 (最小スコア重み) ~ 1000 (最大スコア重み) の整数です。デフォルト値は 100 です。

word-or-phrase

検索する語または句。語の中で使用できる文字は、言語によって違います。語と語の間を区切り文字で区切る必要があるかどうか言語によって違います。英語およびその他の大部分の言語の場合、句の中の語と語の間は空白文字で区切る必要があります。

二重引用符を含む文字ストリングを検索するには、二重引用符を二度入力します。たとえば、テキスト `"wildcard"` 文字を検索するには、次のようにします。

```
""wildcard"" character"
```

この例では、1 組の引用符しか検索できないことに注意してください。1 つの文から 2 組の引用符は検索できません。また、各語または句の最大長は 128 バイトです。

Masking characters

語に次のマスク文字を含めることができます。

_ (下線)

任意の 1 文字を表します。

% (パーセント)

数が不定の任意の文字を表します。単一の % からなる語を指定した場合は、任意の長さのオプションの語を表します。1 つの単語をマスク文字だけにすることはできません。ただし、例外として単一の % でオプションの語を表すことは可能です。マスク文字を使用する場合は、`THESAURUS` を使用できません。マスク文字は、英数字以外の文字の後に続けることはできません。

ESCAPE escape-character

次の文字を検索対象の文字として識別し、マスク文字として使用される文字としては識別しない文字。たとえば、エスケープ文字が \$ であれば、\$%、\$_、および \$\$ は、それぞれ %、_、および \$ を意味します。% および _ 文字の前に \$ がいない場合は、マスク文字を表します。

検索時には、1 バイト文字の使用のみ許可されています。2 バイト文字の使用は許可されていません。

THESAURUS *thesaurus-name*

`text-literal` の拡張に使用されるシソーラスの名前を指定するために使用されるキーワード。シソーラス名は、シソーラス・コンパイラを使用してコンパイルされたシソーラスのファイル名 (拡張子なし) です。

`<os-dependent>/sql1lib/db2ext/thes` に配置されている必要があります。あるいは、パスをファイル名の前に指定することもできます。

EXPAND *relation*

どの関係がシソーラスを使用して `text-literal` の拡張に使用されるかを指定します。シソーラスには、`DB2EXTTH` コマンドで記述された事前定義の関係があります。これらは、次のキーワードを使用して参照されます。

- **SYNONYM**。同等を表す対称のリレーションシップ。
- **RELATED**。関連を表す対称のリレーションシップ。
- **BROADER**。指定された深さレベルを後に続けることができる、指定階層リレーションシップ。
- **NARROWER**。指定された深さレベルを後に続けることができる、指定階層リレーションシップ。

ユーザー定義の関係の場合、`DB2TEXTTH` での関係定義に対応する `RELATION(number)` を使用します。

TERM OF *text-literal*

シソーラスから他の検索項目が追加される `text-literal`。

count **LEVELS**

指定された関係の検索項目を拡張するために使用される、シソーラスでの項目のレベルの数 (深さ) を指定するために使用するキーワード。このキーワードを指定しない場合は、カウント 1 が想定されます。深さの値は、正の整数でなければなりません。

ATTRIBUTE *Attribute-name*

指定された条件に一致する属性を持つ文書を検索します。 `attribute-name` は、`CREATE INDEX` コマンドでの属性表現の名前、または文書モデル・ファイルの属性定義を指します。

属性係数は、ダブルのタイプの属性にのみ許可されます。値の精度は 15 桁が保証されています。16 文字以上の数は丸められます。マスク文字の使用は、`attribute-name`、`valueFrom` および `valueTo` では許可されません。詳細については、次の説明を参照してください。

BETWEEN `valueFrom` **AND** `valueTo`

属性の値が `valueFrom` より大きく (等しくない)、`valueTo` より小さい (等しくない) 場合に、**BETWEEN** 属性係数は真であると評価されます。

>`valueFrom`

属性の値が `valueFrom` より大きい (等しくない) 場合に、**>** 属性係数は真であると評価されます。

<`valueTo`

属性の値が `valueTo` より小さい (等しくない) 場合に、**<** 属性係数は真であると評価されます。

CREATE INDEX コマンドの属性名を引用符を付けて指定するかモデル・ファイルで定義すると、指定した属性名は正確に一致しなければなりません。これに対し CREATE INDEX コマンドで引用符を指定しない場合、属性名は英大文字でなければなりません。

IS ABOUT language word-or-phrase

フリー・テキスト検索引き数を指定できるようにするオプション。異なる種類のスコア・アルゴリズムが使用され、文書内での用語の位置がチェックされます。 word-or-phrase で使用される用語がより近くに集まっているほど、それらの用語はより多くその文書に含まれていると判断され、より高いスコア値が戻されます。

language に許可される値は、215 ページの『付録 E. サポートされている言語』に記載されており、それはタイ語にのみ関係する情報です。指定されない場合は、デフォルトとして en_US が使用されます。language は、word-or-phrase のトークン化のためだけに使用されます。

IS ABOUT が有効なのは、スコア値が要求されており、検索結果がスコア値の順に配列される場合だけであることに注意してください。

第 15 章 SQL スカラー検索関数および SQL 表値関数

DB2 Net Search Extender には、DB2 Universal Database に保管されているテキスト文書を検索する SQL スカラー検索関数および SQL 表値関数が備わっています。

この章では、以下の SQL 検索関数について説明します。

検索関数のサマリー

検索関数	用途	ページ
CONTAINS	特定の文書のテキストを検索する。	156
NUMBEROFMATCHES	検索して、検出された一致の数を返す。	157
SCORE	検索して、検出されたテキスト文書のスコア値を返す。	158
DB2EXT.TEXTSEARCH	SQL 表値関数は検出された主キーの表、一致の数、またはスコア値 (あるいはこれらすべて) を返す。	159
DB2EXT.HIGHLIGHT	その文書が検索結果として判断された理由に関する情報の入手。	163

SQL スカラー検索関数および SQL 表値関数の使用例については、73 ページの『第 8 章 検索』を参照してください。

CONTAINS

CONTAINS スカラー関数は、Net Search Extender によって索引付けされたテキスト文書を検索します。文書にテキストまたは検索引き数に指定された関係が含まれている場合は、INTEGER 値 1 を返します。それ以外の場合は、0 を返します。

関数の構文

▶—CONTAINS—(—*column-name*—,—*search-argument*—)————▶

関数パラメーター

column name

列の名前。列には、関連したテキスト索引が必要です。テキスト索引は、管理コマンド DB2TEXT CREATE INDEX を使用して作成することができます。

search-argument

検索する用語を含む、タイプが VARCHAR のストリング。147 ページの『第 14 章 検索引き数の構文』を参照してください。

注

CONTAINS 照会は、ビューに関して作成されたテキスト索引については使用できません。

NUMBEROFMATCHES

NUMBEROFMATCHES スカラー関数は、テキスト文書内を検索し、文書ごとに一致した結果の数を示す INTEGER 値を戻します。

関数の構文

▶▶—NUMBEROFMATCHES—(—*column-name*—,—*search-argument*—)————▶▶

関数パラメーター

column name

列の名前。列には、関連したテキスト索引が必要です。テキスト索引は、管理コマンド DB2TEXT CREATE INDEX を使用して作成することができます。

search-argument

検索する用語を含む、タイプが VARCHAR のストリング。147 ページの『第 14 章 検索引き数の構文』を参照してください。

注

NUMBEROFMATCHES 照会は、ビューに関して作成されたテキスト索引については使用できません。

SCORE

SCORE スカラー関数は、テキスト文書内を検索して、検出された文書が検索引き数でどの程度よく表されているかを示すスコア値を、検出された文書ごとに戻すことができます。

SCORE は、DOUBLE 値を戻します。検索語がより頻繁に文書で出現するほど、文書のスコアは増加します。

関数の構文

▶▶—SCORE—(—*column-name*—,—*search-argument*—)—————▶▶

関数パラメーター

column name

列の名前。列には、関連したテキスト索引が必要です。テキスト索引は、管理コマンド DB2TEXT CREATE INDEX を使用して作成することができます。

search-argument

検索する用語を含む、タイプが VARCHAR のストリング。147 ページの『第 14 章 検索引き数の構文』を参照してください。

注

SCORE 照会は、ビューに関して作成されたテキスト索引については使用できません。

DB2EXT.TEXTSEARCH

ストアド・プロシージャ検索および SQL スカラー検索関数に加えて、Net Search Extender は、ストアド・プロシージャに極めてよく似た 2 つの SQL 表値関数を備えています。

それらの表値関数は両方とも `db2ext.textsearch` と呼ばれます。2 つの違いは、その一方が `HIGHLIGHT` 関数をサポートし、2 つの追加パラメーター `numberOfHits` と `hitInformation` を持っているということだけです。

コンパウンド主キーを持つ表では、表値関数を使用できないことに注意してください。

`HIGHLIGHT` 関数の使用については、163 ページの『DB2EXT.HIGHLIGHT』を参照してください。

注

この表値関数は、ユーザー表が単一ノード表スペースに保管された場合のみ、分散 DB2 環境によって使用されます。適切なノードに `DB2NODE` 環境変数を使用して接続していることを確認することも必要です。

関数の構文

1. db2ext.textsearch without highlight support

```
db2ext.textSearch
(
  query           VARCHAR(4096),
  indexSchema     VARCHAR(128),
  indexName       VARCHAR(128),
  resultFirstRow  INTEGER,
  resultNumberRows  INTEGER,
  primKeyBinding  <supported types>, // same type as primary key
)

return table
(
  primKey         <supported types>, // same type as primary key
  numberOfMatches INTEGER,
  score           DOUBLE,
  totalNbResults  INTEGER
)
```

2. db2ext.textsearch with highlight support

```
db2ext.textSearch
(
  query           VARCHAR(4096),
  indexSchema     VARCHAR(128),
  indexName       VARCHAR(128),
  resultFirstRow  INTEGER,
  resultNumberRows  INTEGER,
  primKeyBinding  <supported types>, // same type as primary key
  numberOfHits    INTEGER
)
```

```

)

return table
(
  primaryKey          <supported types>, // same type as primary key
  numberOfMatches    INTEGER,
  score               DOUBLE,
  totalNbResults     INTEGER
  hitInformation     BLOB(20K)
)

```

関数パラメーター

入力パラメーターを以下に示します。

query 詳細については、147 ページの『第 14 章 検索引き数の構文』を参照してください。

indexSchema, indexName

検索する索引を識別します。詳細については、120 ページの『CREATE INDEX』を参照してください。

resultFirstRow

照会の結果リストは、分けて戻されます。このパラメーターは、照会結果リストのどの行が、表値関数の結果表に最初に書き込まれるかを記述します。値は 0 以上でなければなりません。

番号 0 は、照会結果リストの最初の行を表わします。

resultNumberRows

このパラメーターは、照会結果リストのうちの何行が表値関数の結果表に書き込まれるかを記述します。0 は、すべての結果が戻されなければならないことを意味します。

このパラメーターは、照会結果リストの最大サイズを決定する結果限界照会パラメーターとは異なることに注意してください。

primaryKeyBinding

このパラメーターのタイプによって、primaryKey Output パラメーターのタイプが決まります。<type1> のタイプの主キーをもった基本表に対して、テキスト索引を作成した場合、primaryKeyBinding も、<type1> のタイプである必要があります。

さらに、このパラメーターによって、テキスト検索の範囲が決まります。primaryKeyBinding を NULL ("CAST(NULL as <type1>") に設定した場合、検索範囲は、索引に保管されているすべての文書になります。あるいは、primaryKeyBinding がバインドされている文書だけに検索を制限することができます。

たとえば、primaryKeyBinding を CAST(5 as BIGINT) に設定する場合、検索を BIGINT 主キー値の「5」を持った単一の文書に制限します。

サポートされているのは、SMALLINT、INTEGER、BIGINT、REAL、DOUBLE、VARCHAR FOR BIT DATA、DATE、TIME、および TIMESTAMP のタイプの、単一の主キーのみであることに注意してください。

numberOfhits

このオプションは、db2ext.textsearch 関数によって戻されるヒット情報の

最大数を指定します。0 を指定すると、最大で 1100 のヒットに関する情報が提供されます。この処理には多大の時間がかかる可能性があります。

このパラメーターは、db2ext.highlight 関数に必要な強調表示情報の組み立てにのみ必要であることに注意してください。

関数パラメーター

以下の戻り値は、さらに結果が要求された場合に正規表に結合する必要のある、一時表に保管されます。SELECT ステートメントで要求されている場合にのみ、NUMBEROFMATCHES、SCORE、TOTALNUMBEROFRESULTS、および HITINFORMATION が計算されることに注意してください。

primKey

検索された文書の主キー。

numberofmatches

NUMBEROFMATCHES は、各文書で一致した個数を示す INTEGER 値です。

score score は、DOUBLE 値を戻します。文書での検索語の頻度が増加すれば、文書スコアが増加します。

totalNumberOfResults

照会結果リストは、検索された結果の個数を示します。各行は同じ値を持つことに注意してください。

また、STOP SEARCH AFTER、または RESULT LIMIT を SCORE 構文とともに照会で使用する場合は、この数がすでに信頼できなくなっていることに注意してください。

hitInformation

db2ext.textsearch から戻されるヒット情報は、強調表示処理に必要です。現在のところ、約 1100 ヒットのヒット情報がこの出力パラメーターに含まれています。ヒット数がこのしきい値を越えた場合は、超過した分のヒット情報が無視されます。

この値は numberOfHits を指定した場合にのみ戻されます。

使用法

SQL 表値関数を使用すれば、ストアード・プロシージャ検索と同じ方法でビュー上を検索できます。ただし、例外として共有メモリーが不要なので、索引をアクティブ化する必要はありません。

この関数は、主としてストアード・プロシージャ検索内で SQL 照会を使用してきたユーザー用です。ただし、基本表上の単一系列の主キーのみがサポートされているという制約事項があります。

次の例は、複数列の主キー表に対してどのように作業できるかを示しています。

```
select s.id from
db2ext.sample s, table (db2ext.textSearch(
    "characteristics",
    'DB2EXT',
    'COMMANDS',
```

SQL 表値関数

```
1,  
20,  
cast(NULL as INTEGER))) t  
where s.id = t.primkey
```

この例では、まずこの表上で単一のユニーク・キーを使用してビューを作成し、次にこのビュー上で索引を作成する必要があります。

SQL 表値関数を `db2ext.highlight` 関数とともに使用する例については、164ページを参照してください。

DB2EXT.HIGHLIGHT

ある文書を検索結果であると判断された理由に関する情報を入手するために、`db2ext.highlight` 関数を使用します。より具体的には、以下の作業を行うためにこの関数を使用できます。

- ヒットの入手
- ヒットおよび周囲のテキストの入手
- ヒットを周囲のユーザー定義の強調表示タグを含む文書の入手

`db2ext.highlight` 関数は `db2ext.textsearch` 表値関数と一緒にしか使用できないことに注意してください。この表値関数は索引を検索し、その結果を `HIGHLIGHT` 関数が使用します。

`db2ext.textsearch` 関数の使用については、159 ページの『`DB2EXT.TEXTSEARCH`』を参照してください。

関数の構文

```
▶—db2ext.highlight—▶
▶-(—document-content—,—hit-information—,—hit-processing-information—)▶
```

関数パラメーター

入力パラメーターを以下に示します。

document content CLOB(100K)

TEXT または XML フォーマットの UTF8 文書だけがサポートされています。この値を大きくするには、109 ページの『`DB2EXTHL` (ユーティリティ)』を参照してください。

hit information BLOB(20K)

ヒット情報を含むストリング。これは、`numberOfHits` パラメーターが指定されている場合に、`db2ext.textsearch` 関数によって戻されます。

hit processing information VARCHAR(1024)

このパラメーターはオプションと値の対のリストです。この対は、" " 文字で囲んだストリング文字をコンマ「,」文字で区切ったものです。指定した文書で強調表示を処理する方法を指定します。オプションを何も指定しないと、元の文書が戻されます。

TAGS = ("STRING", "STRING")

このオプションを使用すると、ユーザーは文書内のヒットの前後に挿入するタグを指定できます。このオプションを省略した場合は、文書内のヒットの前後にタグは追加されません。

WINDOW_NUMBER = INTEGER

このオプションは、強調表示関数が文書の部分 (またはウィンドウ) をいくつ戻すようにするかを指定します。各ウィンドウには 1 つ以上のヒットが含まれています。各ウィンドウの最初のヒットにより、ユーザーに戻される文書の部分が決まります。これらのヒットは、周囲のテキストで囲まれている場合とそうでない場合があります。

このオプションを省略した場合、デフォルトとして 0 を取り、開始および終了タグ (指定されている場合) を含む文書全体が戻されます。このとき、WINDOW_SIZE オプションは無視されます。

WINDOW_SIZE = INTEGER

このオプションはウィンドウの推奨サイズをバイト数で指定します。この実際のサイズは、ヒット数、ヒットの長さ、および開始タグと終了タグのサイズによって異なります。このオプションを省略した場合、0 がデフォルトとなり、周囲のテキストなしでヒットだけが戻されます。

WINDOW_SEPARATOR = "STRING"

このオプションは、1 つのウィンドウと次のウィンドウとを区切るために使用するタグを指定します。このオプションを省略した場合、"..." がデフォルト値になります。

FORMAT = "STRING"

このオプションは文書のフォーマットを指定します。有効な値は XML または TEXT です。このオプションを省略した場合、デフォルト値として TEXT を取ります。フォーマット値が、索引作成時に指定した値と同じになるようにしてください。

MODEL_NAME = "STRING"

このオプションは、指定した XML 文書に関連するモデル名を指定します。FORMAT が TEXT の場合、このオプションを指定するとエラー条件になるので注意してください。

SECTIONS = ("section-name1", ..., "section-nameN")

XML 文書の場合、強調表示を関係のあるセクションに制限することができます。たとえば、そのセクションをモデル・ファイルに定義できます。セクションを指定するには、1 つ以上のセクション名をコンマで区切ります。このオプションを省略した場合、XML 文書全体が強調表示されます。FORMAT が TEXT の場合は、このオプションは無視されるので注意してください。

関数パラメーター

戻りパラメーターを以下に示します。

CLOB(200K)

HIGHLIGHT 関数は、関数自身による文書の変更部分を含む CLOB 値を戻します。

使用法

次の例は、HIGHLIGHT 関数の使い方を示しています。

```
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 = ("section1-name", "section2-name"))'
```

```
FROM patent p, table (db2ext.textsearch(
    'relational database systems',
    'DB2EXT',
    'TI_FOR_CONTENT',
    0,
    20,
    CAST(NULL as BIGINT),
    15)) t
```

```
WHERE p.id = t.primkey
```

100 KB を超える文書を使用すると、SQL 照会が終了し、SQL エラーが発生します (SQL 1476N と sql エラー -433)。これを回避するには、db2exthl コマンドを使用して、文書の内容のサイズを大きくしてください。詳しくは、109 ページの『DB2EXTHL (ユーティリティ)』を参照してください。

注

特殊文字 (「改行」など) は現状のまま戻されます。

制約

- XML およびフラット・テキストの文書のみサポートされています。
- UTF8 のデータベースのみがサポートされています。バイナリーまたはデータ・リンク文書の場合、文書が UTF8 であることを確認する必要があります。
- タイ語の文書はサポートされていません。
- 索引付けのときと照会のときとで使用した文書フォーマットが同じでない場合、HIGHLIGHT 関数から予測不能な結果が戻されます。
- 文書のテキスト部分で検出されたヒットだけが強調表示されます。
- 強調表示関数は db2ext.textsearch 関数と一緒にしか使用できません。
- ストリング値には " 文字を含めることはできません。

第 16 章 ストアード・プロシージャ検索関数

Net Search Extender は、事前定義の結果表を戻すストアード・プロシージャ検索を備えています。結果表は、索引の作成中に CACHE TABLE セクションで指定されます。少ない数の結果を特定の順序で戻す場合には、ストアード・プロシージャ検索を使用します。

たとえば、インターネット・アプリケーションで、最初の 20 行を戻し、残りの結果も 20 行ずつ戻したい場合などがあげられます。

注

このストアード・プロシージャ関数は、ユーザー表が単一ノード表スペースに保管された場合にのみ、分散 DB2 環境によって使用されます。

適切なノードに DB2NODE 環境変数を使用して接続していることを確認することも必要です。

DB2EXT.TEXTSEARCH (ストアド・プロシージャ検索用)

関数の構文

```
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 )
```

関数パラメーター

入力パラメーターを以下に示します。

Query 詳しくは、147 ページの『第 14 章 検索引き数の構文』を参照してください。

indexSchema, indexName

検索する索引の識別。 120 ページの『CREATE INDEX』を参照してください。

resultFirstrow

照会結果リストは、部分的に戻されます。パラメーターは、照会結果リストのどの行が、ストアド・プロシージャの結果セットに最初に書き込まれるかを記述します。照会結果リストの最初の行は、番号 0 で識別されます。

resultNumberRows

このパラメーターは、照会結果リストのうちの何行がストアド・プロシージャの結果セットに書き込まれるかを記述します。

これを、照会結果リストの最大サイズを判別する、照会の "result limit" 式と混同してはなりません。

値は 0 以上でなければなりません。0 の場合は、すべての結果を戻す必要のあることを意味します。

注

大きな結果セットを要求する場合は、一時ユーザー表スペースを使用可能にしてください。使用できるものがない場合は、表スペースを作成してください。以下の例は、UNIX プラットフォーム上で表スペースを作成しています。

```
db2 "create user temporary tablespace tempts managed by system
    using ('/work/tempt.ts')"
```

scoringFlag

0 は、スコアがないことを意味し、1 はスコアがあることを意味します。スコアリングが要求されている場合に、追加の列に値の高い順でスコア値が入れられて戻されます。

searchTermCountsFlag

このパラメーターは searchTermCounts 処理を制御します。searchTermCountsFlag が 0 のときは、searchTermCounts は計算されません。

関数パラメーター

出力パラメーターを以下に示します。

searchTermCounts

索引内の検索項目照会ごとの出現数。これらのカウントは、照会の検索項目と同じ順序でブランクで区切られたリストとして戻されます。

詳細については、**searchTermCountsFlag** を参照してください。

totalNumberOfResults

照会結果リストで検出された結果の合計数。

また、STOP SEARCH AFTER、または RESULT LIMIT を scoringFlag 構文とともに照会を使用する場合は、この数がすでに信頼できなくなっていることに注意してください。

使用法

ストアード・プロシージャによって戻された結果セットの列は、DB2TEXT CREATE INDEX コマンドの CACHE TABLE オプションにより表示されます。scoringFlag=1 の場合は、タイプがダブルの列が追加されます。この列には、SCORE 値が入っています。

最初の照会と同じストリングが指定されている 2 番目の照会のパフォーマンスを向上させるには、以下のオプションを使用します。ただし、totalNumberOfResults が不要な別のカーソル・ウィンドウで作業しなければならないことに注意してください。

- スコアリングが必要ない場合は、次の構文を追加します。STOP SEARCH AFTER x DOCUMENTS (ここで、x の値は resultFirstRow + resultNumberRows です)。
- スコアリングが必要である場合は、次の構文を追加します。STOP SEARCH AFTER y DOCUMENTS (ここで、y の値は最初の照会の totalNumberOfResults と同じです)。

検索のために適切なノードと接続するには、DB2NODE 環境変数を設定する必要があるかもしれません。

UNIX の場合には、次のコマンドを使用してください。

```
export DB2NODE=<no>
```

すべての物理ノードの時刻が同期していることが重要であることに注意してください。

Windows の場合には、次のコマンドを実行します。

ストアード・プロシージャ検索関数

```
set DB2NODE= <no>
```

注

インスタンス所有者 ID と異なる fenced ユーザー ID では、パーティション・データベースを扱う作業ができません。

第 17 章 構造化文書のサポート

構造化文書は、文書モデルと文書ファイル定義から構成されています。

文書モデルが構造化文書を記述する方法

HTML や XML 形式の文書は構造化文書の例です。これらの文書には、テキスト・フィールドや文書属性を識別するタグが含まれています。テキスト・フィールドには、文書の表題、作成者、または文書の説明などの情報を含めることができます。文書属性には数を含めることができます。

以下に示すのは、構造化されたプレーン・テキスト文書の一部です。HTML のようなタグで区切られたフィールドと属性があります。

```
[head]Handling structured documents
[/head]

[abstract]This document describes the concept of structured documents
and the use of document models to...
[/abstract]
:
:
```

構造化文書の索引を作成する場合、Net Search Extender はその構造を認識していなければなりません。構造を認識することによって、テキスト・フィールドと属性に索引を付けることができ、またフィールドや属性を識別するユニーク名を指定して、それらをまとめて保管することもできます。こうして、Net Search Extender を使用して、特定のテキスト・フィールドを選択的に検索したり、特定の属性を持つ文書を検索したりすることが可能になります。

Net Search Extender が特定の文書フォーマットを理解できるようにするには、文書モデル内の構造の定義を Net Search Extender に渡す必要があります。

CREATE INDEX コマンドを呼び出して文書に索引を付けるときに、文書のモデルの名前を引数として指定します。

文書モデルを使用して文書に索引を付ける前に、まず文書モデルを定義してから、索引に文書モデルを追加する必要があります。

文書モデルの例

索引を付ける文書フォーマットごとに 1 つの文書モデルを定義する必要があります。以下は、プレーン・テキスト構造化文書の簡単な文書モデルです。この例の中の GPP は General Purpose Parser (汎用パーサー) の略です。

```
<?xml version="1.0"?>
<GPPModel>                                - here begins the GPP document model

  <GPPFieldDefinition                       - here begins a field definition
    name="Head"                             - the name you assign to this field
    start="[head]"                         - the boundary string at the beginning of the field
    end="[/head]"                          - the boundary string at the end of the field
    exclude="YES" />
```

```
<GPPFieldDefinition      - here begins the next field definition
name="Abstract"
start="[abstract]"
end="[/abstract]"
exclude="NO" />
:
:
</GPPModel>
```

文書モデルは、239 ページの『付録 G. 文書モデルの参照情報』に定義されているタグを使用して XML 言語で指定します。文書モデルはテキスト・フィールド定義と属性定義からなっています。上記の例に示されているのは、GPPFieldDefinition エレメントに定義されているテキスト・フィールド定義のみです。同じようにして、GPPAttributeDefinition を使用して文書属性を定義できます。

1 行目の `<?xml version="1.0"?>` は、文書モデルが XML タグを使用して書かれていることを示しています。テキスト・フィールド定義ごとに、フィールド定義の開始 (start) と終了 (end) を表す境界ストリングをソース・ドキュメントに指定します。したがって、ある文書に `[head]`、何かのテキスト、`[/head]` という順序で続く文字シーケンスが含まれる場合は、それらの境界ストリングの間のテキストが `head` という名前を割り当てられたテキスト・フィールドの内容であると解釈されます。

各フィールド定義にはフィールド名を割り当てます。このフィールド名を使うことにより、照会はテキスト・フィールドの内容の検索を制限できます。フィールド名は、決まった名前を指定することも、またはある規則によって構造単位の内容から取った名前にすることもできます。たとえば、XML エンティティーのタグ名や XML 属性の名前などをそのようにして決めることができます。

文書モデル

文書モデルは、主として文書の構造のどの部分に索引を付ける必要があるか、どのように索引を付けるかを制御します。その目的は次のとおりです。

- ソース・ドキュメントで区別する必要があるテキスト・フィールドを識別する
- 上記のようなテキスト・フィールドのタイプを判別する
- フィールド名をテキスト・フィールドに割り当てる

文書モデルがテキストをテキスト・フィールドに属していると識別すると、テキストは文書のテキストの内容の一部であると見なされ、用語が抽出されて索引に保管されます。

文書モデルのエレメントは、その文書フォーマットに使用されたパーサーによって異なります。

- HTML フォーマットの場合、文書モデルは、HTML タグ名を使用して、どのタグに索引を付けるべきか、メタタグ情報をどのように処理するかを定義します。
- XML フォーマットの場合、定義済みの一連のタグは存在しないので、文書モデルはまず、どのタグが目的のタグであるかを定義する必要があります。同じ名前の XML エレメントも、他のどのエレメントに組み込まれているかに基づいて区別できます。

- GPP (汎用パーサー) フォーマットの場合、文書モデルはテキスト・フィールドの境界を判別する必要があるために、より深くパーサーと相互作用します。ここではフィールド定義が、フィールドの境界を検出するためにストリングを指定する必要があります。
- Outside-In フォーマットの場合、文書モデルは、HTML タグ名に類似したタグを使用して、どのタグに索引を付けるべきか、メタタグ情報をどのように処理するかを定義します。Outside-In フィルター・フォーマットを INSO というのもあるので注意してください。

詳細に関しては『文書モデルの定義』のセクションを参照してください。

文書タイプ定義 (DTD) の形式での文書モデルの構文、およびテキスト・フィールドの制限については、239 ページの『付録 G. 文書モデルの参照情報』を参照してください。

テキスト・フィールドおよび文書属性

文書モデルを使用することにより、文書の特定の部分またはセクションを、特定のテキスト・フィールドに属するもの、または文書属性、あるいはその両方として識別できます。

テキスト・フィールド

文書のテキストは、テキスト・フィールドの一部であるかどうかに関係無く、完全に索引付けされます。文書モデルがテキストをテキスト・フィールドに属していると識別すると、そのテキストは文書のテキストの内容の一部であると見なされ、通常の方法で索引が付けられます。つまり、意味のある用語は抽出されて索引に保管されます。したがって、テキスト検索に制約がない場合、そのテキストは検索の対象となります。

このように、特定のフィールドのテキストを選択的に検索することが可能です。たとえば、structure というワードをテキスト・フィールド Abstract に持つ文書を検索することができます。

1 つの文書に同じテキスト・フィールドが複数回出現してもかまいません。たとえば、すべての図のキャプションに適用されるテキスト・フィールドを定義することができます。あるテキスト・フィールドが別のテキスト・フィールドにオーバーラップすることもあります。

特定のテキスト・フィールドの内容に索引付けをしないようにしたい場合は、exclude="YES" を含むフィールド定義を指定できます。テキスト・フィールドおよび文書属性の制限のリストが、242 ページの『テキスト・フィールドおよび文書属性の制限』に示されています。

文書属性

文書属性には、number というタイプの短い定様式の情報が組み込まれています。この属性を含む文書の検索には、値の範囲を使用できます。

属性は索引テキストとともに保管されず、別個の項目索引に保管されます。そのため、属性の内容で文書を検索するには、属性を明示的に指定した属性検索を行う必要があります。

名前属性: Net Search Extender には浮動小数点数を認識するパーサーが備わっています。整数値のフォーマットの正しい例と誤った例を以下に示します。

表6. サポートされる文書フォーマットに対するデフォルトの文書モデルの動作

正しいフォーマット	誤ったフォーマット
1000 1 000 1.000 - ピリオドは小数点文字	1,000
100 000 100 000.00123	1 000 000 - 1 と 0 の間に スペースが 2 つ

数値の小数部にはスペース文字を使用することができないことに注意してください。たとえば、1 000.000 100 は 2 つの数 1000.000 と 100 であるものとして扱われます。

言語固有の区切り記号および言語固有の通貨フォーマットはサポートされていません。

デフォルトの文書モデル

HTML、XML、および Outside-In フィルター文書の場合、Net Search Extender は文書モデルを定義しない場合に使用されるデフォルトの文書モデルが用意されています。構造化プレーン・テキスト文書の場合、文書モデルを指定する必要があります。

デフォルトの文書モデルの 1 つを使用する場合は、

- すべてのフィールドに索引が付けられ、メタ情報などの特殊な情報は抽出されません。
 - HTML および INSO フォーマットの場合、各フィールドには対応するタグの名前が割り当てられます。
 - XML の場合、XML 文書のすべての XML ノードは、各ノードの完全修飾タグ・パスによって識別される重複フィールドにマップされます。たとえば、パス /play/role/name です。
- 数値属性の索引は作成されません (数値属性がデフォルトの文書モデルでは定義されていないためです)。

表7. サポートされる文書フォーマットに対するデフォルトの文書モデルの動作

文書タイプ	デフォルトの文書モデルの動作
HTML	次のタグをテキスト・フィールドとして受け入れる: <a> <address> <au> <author> <h1> <h2> <h3> <h4> <h5> <h6> <title>。 フィールド名は、タグ名。例: "address"。
XML	すべてのタグをテキスト・フィールドとして受け入れる。 フィールド名は、完全修飾タグ・パス名。例: "/play/title"。
構造化プレーン・テキスト (GPP)	デフォルトの文書モデルなし。

表 7. サポートされる文書フォーマットに対するデフォルトの文書モデルの動作 (続き)

文書タイプ	デフォルトの文書モデルの動作
Outside-In (INSO)	Outside-In フィルターによって戻されるものとして 184 ページの『エレメント・パラメーター』に示されている文書プロパティをテキスト・フィールドとして受け入れる。フィールド名は Outside-In が使用する文書プロパティの名前です。 例: "SCCCA_TITLE"。

文書のタイプごとに文書モデルが定義されます。モデルはすべて異なるので、例と説明はそれぞれのモデルごとに記載されています。

注

デフォルトの文書モデルでも正しく文書进行处理しますが、索引付けと検索の効率を上げるには、独自の文書モデルを定義する必要があります。

デフォルトの文書モデルを使用すると、文書のテキストは、テキスト・フィールドの一部であるかどうかに関係無く、完全に索引付けされます。したがって、テキスト検索に制約がない場合、そのテキストは検索の対象となります。

構造化プレーン・テキストの場合の文書モデルの定義

以下に汎用パーサー (GPP) 構造化プレーン・テキスト文書の例を示します。

```
[head]Handling structured documents
[/head]
[year]2002
[/year]
[abstract]This document describes the concept of structured documents
and the use of document models to...
[/abstract]
```

以下に GPP 文書モデルの例を示します。

```
<?xml version="1.0"?>
<GPPModel>

  <GPPFieldDefinition
    name="Head"
    start="[head]"
    end="[/head]"
    exclude="YES" />

  <GPPFieldDefinition
    name="Abstract"
    start="[abstract]"
    end="[/abstract]"
    exclude="NO" />
    - This is the start of text field
    - This is the end of a text field

  <GPPAttributeDefinition
    name="year"
    start="[year]"
    end="[/year]"
    type="NUMBER" />
    - This is the start of a document
      attribute
    - This is the end of a document
      attribute

</GPPModel>
```

1 行目の `<?xml version="1.0"?>` は、文書モデルが XML タグを使用して書かれていることを示しています。このモデルは XML フォーマット文書用には書かれていないことに注意してください。

各フィールドは、エレメント・パラメーターを含む `GPPFieldDefinition` タグまたは `GPPAttributeDefinition` タグ内で定義されます。

すべての定義は、`<GPPModel>` タグ内に含まれる必要があります。

エレメント・パラメーター

文書モデル・エレメントのパラメーターは下記のとおりです。

name 定義ごとに名前をテキスト・フィールドまたは文書属性に割り当てます。この名前を使用すれば、検索照会を特定のテキスト・フィールドまたは文書属性の内容に制限できます。上記の例を使用すると、`Abstract` という名前の付いたテキスト・フィールド内で、`structure` という単語を含んだ文書を検索できます。

start テキスト・フィールドまたは文書属性の開始をマークするコード・ページ UTF-8 の境界ストリング。境界ストリングの指定に規則はありません。任意の UTF-8 ストリングとすることができます。たとえば、次のような例が挙げられます。start="introduction:"、start="note!"、start="\$\$...."

印刷できない文字および特殊な XML 文字 "<" と "&" は、標準 XML エスケープ文字 ("<" の場合、"<"、"&" の場合 "&") を使用して指定する必要があります。

end オプション。テキスト・フィールドまたは文書属性の終了をマークするコード・ページ UTF-8 の境界ストリング。終了タグを指定しない場合、次に検出された開始タグは、フィールドの終了と見なされます。これ以降、開始タグが検出されない場合は、フィールドは文書の終わりまで拡張され、これ以上のフィールドは識別されません。

type 文書属性のタイプは常に「NUMBER」でなければなりません。このパラメーターは、フィールド定義には適用されません。

exclude

YES または NO。フィールド定義内のテキストを除外して索引を付けないかどうかを決定するパラメーター。このパラメーターは、属性定義には適用されません。

この例では、フィールド定義「head」は除外されますが、定義「abstract」は含まれます。

制約事項：

- 同じ開始タグを持つ 2 つのフィールド定義または属性定義があってはなりません。ただし、フィールド定義および属性定義は、同じ開始タグと終了タグをもつことはできます。
- 開始タグは別のタグに固有の接頭部であってはなりません。たとえば、開始タグ "author" と開始タグ "authority" を指定することはできません。
- 開始タグおよび終了タグは空ストリングであってはなりません。

文書タイプ定義については、239 ページの『文書モデル用の DTD』を参照してください。

追加の制約事項については、242 ページの『テキスト・フィールドおよび文書属性の制限』を参照してください。

GPP 文書を索引付けするとどうなるか

汎用パーサーは、開始境界ストリングの 1 つを検索して文書をスキャンします。開始境界ストリングを検出すると、対応する終了境界ストリングを検出するまで、後続のフィールドを解析します。

次に、定義用語に従って、つまりテキスト・フィールドまたは文書属性として、フィールドの内容に索引が付けられます。テキスト・フィールドと文書属性が同じ開始境界ストリングと終了境界ストリングをもつ場合、フィールドの内容には、テキスト・フィールドおよび文書属性の両方として索引が付けられます。

フィールドのネスティングは許可されていません。終了境界ストリングに到達する前に新たな開始境界ストリングが検出されると、新たな開始境界ストリングは通常のテキストとして解釈されます。

対応する終了境界ストリングが検出されない場合、フィールドは文書の終わりまで拡張していると想定されます。その場合は適切な理由コードが報告されます。

文書モデルで終了境界ストリングが指定されない場合、新規の開始境界ストリングが直前のフィールドの終了のシグナルを出します。

HTML 文書の場合の文書モデルの定義

HTML パーサーはテキストを UTF-8 コード・ページに変換します。その結果、HTML タグを認識し、タグ・クラスに分類します。

- 無視されるタグ付き情報。例：フォント情報など。
- 定位置情報を提供するタグ。例：新規の段落用の `<p>`; など。
- 構造上の情報を提供するタグ。例：`<Title>` など。

このパーサーは、HTML 4 で定義されたすべての文字エンティティ参照 (例：`ä`; (ä) など) を認識し、UTF-8 での対応するコード・ポイントに分解します。

メタ・タグを認識し、メタ・タグ・テキストを解析します。

以下に HTML 文書の例を示します。

```
<HTML>
<HEAD>
<META NAME="year" CONTENT="2002">
<TITLE> The Firm </TITLE>
</HEAD>
<BODY>
<H1>Synopsis</H1>;
```

```
<H1>Prologue</H1>;:  
:  
</BODY>
```

以下に HTML 文書モデルの例を示します。

```
<?xml version="1.0"?>  
<HTMLModel>  
  
  <HTMLFieldDefinition  
    name="subtitle"  
    tag="title"  
    exclude="YES" />  
  
  <HTMLFieldDefinition  
    name="header1"           - This is the start of text field  
    tag="h1"  
    exclude="YES" />       - This is the end of the text field  
  
  <HTMLAttributeDefinition  
    name="year"              - This is the start of the document  
    tag="meta"               attribute  
    meta-qualifier="year"  
    type="NUMBER" />       - This is the end of the document  
                               attribute  
  
</HTMLModel>
```

1 行目の `<?xml version="1.0"?>` は、文書モデルが XML タグを使用して書かれていることを示しています。このモデルは XML フォーマット文書用には書かれていないことに注意してください。

各フィールドは、エレメント・パラメーターを含む `HTMLFieldDefinition` タグまたは `HTMLAttributeDefinition` タグ内で定義されます。

すべてのテキスト・フィールド定義は、`<HTMLModel>` タグ内に含まれる必要があります。

エレメント・パラメーター

文書モデル・エレメントのパラメーターは下記のとおりです。

name 定義ごとに名前をテキスト・フィールドまたは文書属性に割り当てます。この名前を使用すれば、検索照会を特定のテキスト・フィールドまたは文書属性の内容に制限できます。上記の例を使用すると、`subtitle` という名前の付いたテキスト・フィールド内で、`firm` という単語を含んだ文書を検索できます。

tag 開始タグおよび (暗黙の) 終了タグがテキスト・フィールドまたは文書属性をマークするエレメントを識別します。その名前のエレメント内にあるテキストは、定義されたフィールドの内容を構成しています。

タグの大文字小文字は無視されます。

上記の例を使用すると、`H1` タグの後に続くテキストには、フィールド「`header1`」の一部としての索引が付けられます。この場合は、「`synopsis`」および「`prologue`」の索引が作成されます。

meta-qualifier

このタグは、**タグ・エレメント**と一緒に使用する必要があります。
「meta」タグを指定することによって、**meta-qualifier** に一致する内容の値が抽出されます。

HTML 文書の例では、**メタ・タグ**は次のエレメントを持ちます。

```
<META NAME="Author" CONTENT="J. Grisham">
```

文書モデルの例では、**meta-qualifier** は「author」です。したがって、**ストリング属性**「author」の値として、内容「J. Grisham」の索引が作成されます。

type 文書属性タイプは「NUMBER」でなければなりません。このパラメーターは、**フィールド定義**には適用されません。

exclude YES または NO。フィールド定義内のテキストを除外して索引を付けないかどうかを決定するパラメーター。このパラメーターは、**属性定義**には適用されません。

この例では、**フィールド定義**「header1」は除外されますが、**定義**「subtitle」は含まれます。

文書のこれ以外のテキストはすべて索引付けされますが、**フィールドの一部**としてではありません。

文書タイプ定義については、239 ページの『**文書モデル用の DTD**』を参照してください。

制約事項については、242 ページの『**テキスト・フィールドおよび文書属性の制限**』を参照してください。

XML 文書の場合の文書モデルの定義

XML 文書の文書モデルを使用することにより、XML 文書内にあるノードをフィールド、文書属性、またはその両方にマップする方法を定義できます。

以下に XML 文書の例を示します。

```
<?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">
    <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>
```

以下に XML 文書モデルの例を示します。

```

<?xml version="1.0"?>
<XMLModel>

<XMLFieldDefinition [1]
name="addresses"
locator="/purchaseOrder/shipAddress"

<XMLFieldDefinition [2]
name="customerName"
locator="//shipAddress/name"

<XMLAttributeDefinition [3]
name="partNumber"
type="NUMBER"
locator="/purchaseOrder//item/@partNo" />

<XMLFieldDefinition [4]
name="none"
locator="/purchaseOrder/@orderDate"
exclude="yes" />

</XMLModel>

```

1 行目の `<?xml version="1.0"?>` は、このモデルが XML を使用して書かれていることを示しています。各フィールドは、エレメント・パラメーターを含む `XMLFieldDefinition` または `XMLAttributeDefinition` タグ内で定義されます。

すべてのテキスト・フィールド定義は、`<XMLModel>` タグ内に含まれる必要があることに注意してください。制約事項については、242 ページの『テキスト・フィールドおよび文書属性の制限』を参照してください。

エレメント・パラメーター

文書モデル・エレメントのパラメーターは下記のとおりです。

name 定義ごとに名前をテキスト・フィールドまたは文書属性に割り当てます。これらの名前を使用すれば、検索照会を特定のテキスト・フィールドまたは文書属性の内容に制限できます。

名前の中で次の変数の 1 つを使用できます。変数は、ソース・ドキュメントの一致するエレメントから生成されたストリングによって置き換えられます。

変数	値
\$(NAME)	XPath に一致する XML エレメントの実際の修飾名 (QName)。
\$(LOCALNAME)	XPath に一致する XML エレメントの実際のローカル名 (接頭部なし)。
\$(PATH)	XPath に一致する XML エレメントのスラッシュおよびタグのシーケンスとしての実際の絶対パス。

type 文書属性タイプは「NUMBER」でなければなりません。このパラメーターは、フィールド定義には適用されません。

locator 検索フィールドとして使用されるソース・ドキュメントの一部を選択する XPath 言語の式。

XML 文書モデル・ファイルの作成時には、ロケータ内での修飾名 (QNames と呼ばれる) が XML 文書内の特定のタグと同一でなければなりません。同一でないと、フィールドが全く認識されないため、フィールドを照会しても結果が戻されません。

ロケータを以下に示します。詳細については、240 ページの『ロケータ (XPath) 式のセマンティクス』の構文を参照してください。

purchaseOrder salesOrder	すべての purchaseOrder エlement および salesOrder エlement
shipAddress	すべての shipAddress エlement
*	すべての Element (これは child::* の省略形 - 詳細については構文を参照してください)
name/item	name の親を持つすべての item エlement
purchaseOrder//item	purchaseOrder を発生源とするすべての item エlement
/	ルート・ノード
comment()	すべてのコメント・ノード
processing-instruction()	すべての処理命令
attribute::* (or @*)	すべての属性ノード

リテラルは、単一または二重引用符で囲まれる文字列です。端末トークンの正確な定義に関しては、XML 推奨を参照してください。

XPath ロケータは、XML Stylesheet Language Transformation (XSLT) パターンに類似しています。これらは、述部、関数 'id' および 'key'、あるいはノード・テスト 'text()' および 'node()' を含まない XSLT パターンのサブセットのみから構成されています。

ignore

YES または NO。パラメータを使用して、ロケータに例外を指定します。

時には * などの汎用ロケータを指定して、索引付けしようとするノードを一致させたい場合があります。しかし、より具体的なロケータに一致する一部のノードを索引付けさせないように指定することもできます。

これを公式化するには、索引付け時に無視するノードに、より具体的なロケータを指定してフィールド定義を組み込みます。こうすることで、このロケータに汎用ロケータを指定したものよりも高い優先順位を与え、ignore="yes" を指定します。これは、一致するノードに対してフィールド情報を生成してはならないことをインデクサーに指示します。

このような無視されたノードが索引作成者生成ノードに組み込まれた場合でも、無視されたノードの内容は、フィールド生成ノードの内容にも属するので、索引付けされることに注意してください。

priority

特定のロケーターによって検出された定義に与えられる優先順位を指定する -1 ~ +1 の間の実数。

優先順位を指定しない場合は、デフォルトの優先順位が使用されます。

- | によって区切られた複数の選択肢は、一連の定義 (各選択肢ごとに 1 つの定義) として処理されます。
- 単一の名前によって一致するロケーター。つまり、次のいずれかの書式を持つロケーターは、デフォルトの優先順位が 0 になります。
 - ChildOrAttributeAxisSpecifier QName
 - ChildOrAttributeAxisSpecifier processing-instruction(Literal))
- ChildOrAttributeAxisSpecifier NCName:* の書式を持つロケーターは、デフォルトの優先順位が -0.25 になります。
- ChildOrAttributeAxisSpecifier NodeTest の書式を持つロケーターは、デフォルトの優先順位が -0.5 になります。
- これ以外のロケーターは、デフォルトの優先順位が 0.5 になります。

ロケーターがより具体的であるほど、デフォルトの優先順位も高くなることに注意してください。たとえば、具体的でないロケーター * が、定義内で低い優先順位を与えられるのに対して、名前はより具体的なロケーターであり、より高い優先順位が与えられます。

また、あるノードが複数のロケーターと一致した場合、優先順位を割り当てることによって、どの定義が選択されたかを判別できることにも注意してください。優先順位が最も高い定義が選択されます。2 つの定義が同じ優先順位であった場合、最新のものが選択されます。

この競合解決は、XML Stylesheet Language Transformation (XSLT) で使用されるものと同一です。

exclude

YES または NO。フィールド定義内のテキストを除外して索引を付けないかどうかを決定するパラメーター。このパラメーターは、属性定義には適用されません。

この例では、フィールド定義「customerName」は除外されますが、定義「addresses」は含まれます。

制約事項：

- フィールドのネストは許可されています。たとえば、ある指定でロケーション XPath に選択されているノードが、別の属性定義によって選択されている XML エlement内にある場合です。しかし、ロケーター式のマッチング・セマンティック上、同じ 1 つの XML ノードを複数のフィールドにマップすることはできません。
- Net Search Extender は、XML 文書のコード・ページを探すことはしません。CREATE INDEX の際に指定された CCSID、あるいは指定されていない場合は、DB2 コード・ページを使用します。

XML 文書を索引付けするとどうなるか

次の表は、索引に書き込まれる内容を示しています。

表 8. テキスト索引の項目

フィールド境界情報	索引付けされるテキスト	
"addresses" フィールドの開始、"customerName" フィールドの開始	Alice Smith	[1] および [2]
"customerName" フィールドの終了	123 Maple Street Mill Hill CA 90999	[2]
"addresses" フィールドの終了	123 1 S&B Lawnmower Type ABC-x 239.90 2001-01-25 987Z 1 Multifunction Rake ZYX 69.90 2001-01-24	
属性名	属性値	
パーツ・ナンバー	123, 987	

この例に示されているように、ネスト・フィールドが可能であることを注意してください。フィールド `addresses` は、XML 文書内でフィールド `customerName` が選択するノードの上位ノードを選択します。したがって、論理上、組み込みノードの内容は両方のフィールドに属します。テキスト・フィールドが重複することはあっても、それらのフィールド内のテキストが索引付けされるのは一回だけです。この例では、フィールド制限を指定して検索を実行すると、`Alice Smith` が `addresses` だけでなく `customerName` にも検出されます。

フィールドの内容は、次の規則に従って決定されます。

- ロケーターがコメント、処理命令、または XML 属性に一致するフィールドの場合、フィールドの内容は実際のコメント・テキスト、処理命令テキスト、または属性値テキストです。
- XML エレメントまたはルート・ノードに一致するフィールドの場合、`exclude="YES"` が指定されたフィールドによって一致するエレメントの場合を除き、フィールドの内容は任意の組み込まれたエレメントからのテキストで構成されています。

文書は適格な XML で構成される必要がありますが、必ずしも DTD を XML 文書で指定する必要はありません。DTD の妥当性検査または外部エンティティー解決は実行されません。Net Search Extender は XML 文書を文書モデルに一致させるだけです。内部エンティティーは XML の要求どおりに置換されます。

文書タイプ定義については、239 ページの『文書モデル用の DTD』を参照してください。

制約事項については、242 ページの『テキスト・フィールドおよび文書属性の制限』を参照してください。

Outside-In フィルター文書の場合の文書モデルの定義

Outside-In フォーマットの文書モデルは、特定のタグ・セットによって識別される構造エレメントを NSE テキスト・フィールドおよび文書属性にマップできるという点で、HTML 文書モデルに非常によく似ています。たとえば、Microsoft Word 文書のセットを持っていて、文書プロパティの "title"、"subject"、および "keyword" をフィールドとして、文書プロパティ "author" および "category" を文書属性として索引付けするとします。次の例の Outside-In 文書モデルを使えば、このマッピングが行えます。

```
<?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>
```

エレメント・パラメーター

文書モデル・エレメントのパラメーターは下記のとおりです。

name テキスト・フィールドまたは文書属性に割り当てる名前。各フィールド定義にはフィールド名を、各属性定義には属性名を割り当てます。これらの名前を使うことにより、照会は特定のテキスト・フィールドの内容に検索を制限したり、特定の属性を持つ文書を検索したりできます。

tag タグを識別します。このタグの開始および終了 (または暗黙終了) エレメントによって、テキスト・フィールドや文書属性が明示されます。指定された名前のエレメント内にあるテキストは、定義されるフィールドまたは属性の内容を構成します。タグの大文字小文字は無視されます。可能な値を以下に説明します。

type 文書属性のタイプは "NUMBER"、"DATE"、または "STRING" のいずれかにすることができます。このパラメーターは、フィールド定義には適用されません。

exclude YES または NO。フィールド定義内のテキストを除外して索引を付けないかどうかを決定するパラメーター。このパラメーターは、属性定義には適用されません。

Outside-In 文書モデルは、それぞれが名前とタグを定義するフィールド定義または属性定義 (あるいはその両方) から構成されています。属性定義の場合はタイプも必須ですが、フィールド定義にはオプションの "exclude" フラグがあります。HTML モデルの場合と同じように、このような定義の名前属性は文書部分のマップ先となる Net Search Extender フィールドまたは属性を定義します。これは、任意の UTF-8 テキスト・ストリングにすることもできます。追加情報については、Outside-In Content Access Specification, Version 7.5 を参照してください。

Outside-In 開始、終了および文書プロパティのタグに関連するタグ属性が取り得る値のリストについての詳細は、243 ページの『Outside-In タグ属性値』を参照してください。

Outside-In 文書を索引付けするとどうなるか

デフォルトでは、すべてのテキストはどのフィールドにも属さないものとして索引付けされます。テキストのストリームの中で現れた開始タグが、現在アクティブである文書モデルのある定義項目と一致すると、開始タグとそれに対応する終了タグの間にあるテキストは、その定義項目に従って処理されます。たとえば、索引付けされたフィールドとして、除外されたフィールドとして、属性として、またはそれらの複合として処理されます。

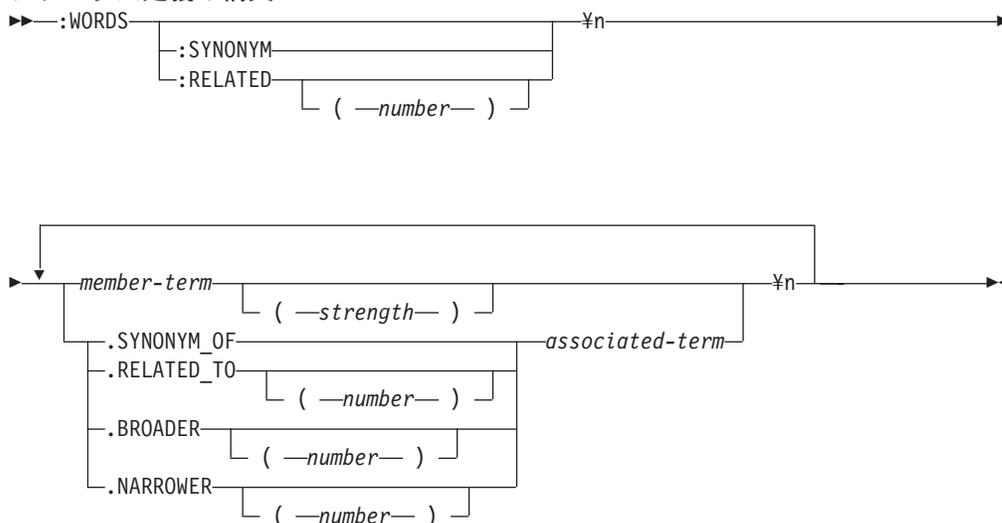
一致する定義がない場合は、開始タグとそれに対応する終了タグが無視されます。

Outside-In フィルターは文書のフォーマットとコード・ページを自動的に認識するので、CCSID 仕様には影響しません。Outside-In フィルターがフォーマットとコード・ページを正確に判別できない場合、文書は ASCII ファイルとして処理されます。

第 18 章 シソーラス・サポート

以下に各定義グループの構文を示します。

シソーラス定義の構文



¥n は構文の一部ではありませんが、シソーラス定義ファイルでの行の終了を示します。

シソーラス定義ファイルでは、以下のようにしてコメント行を挿入できます。

```
# my comment text
```

:WORDS

関連するワード・グループを開始するキーワード。

:SYNONYM,

:RELATED [(number)],

関係名。

関係名は関係タイプおよび番号から構成されます。番号を省略すると、システム提供の関係名であるゼロが想定されます。:SYNONYM は常にシステム提供の関係名です。

:SYNONYM などのコロンで始まる関係名は、同じ関係で互いに関連しているワード・リストの前に置きます。例：

```
:WORDS
:SYNONYM
  air steward
  cabin staff member
  flight attendant
```

member-term

シソーラス辞書に組み込む用語。

- 最大長は 64 バイトです (コード・ページ UTF-8 の場合は 42 バイト)。
- 同一文字の 1 バイト文字と 2 バイト文字は、同じと見なされます。

- 英大文字および小文字は区別されません。
- 用語にはブランク文字を含めることができます。
- 単一バイト文字のピリオド "." またはコロン ":" は使用できません。

このパラメーターは、検索した用語と弱い関係をもつワードをシソーラス検索に含ませたくない場合に役に立ちます。強さは、1 から 100 までの数値です。デフォルト値は 100 です。

**.SYNONYM_OF,
.RELATED_TO [(number)], .BROADER [(number)],
.NARROWER [(number)]**

関係名。関係名は関係タイプおよび番号から構成されます。番号を省略すると、システム提供の関係名であるゼロが想定されます。関係名 .SYNONYM は常にシステム提供の関係名です。

.SYNONYM_OF などのピリオドで始まる関係名は、あるワードと別のワードとの関係を定義します。例：

```
:WORDS
  air steward
  .SYNONYM_OF cabin staff member
  .SYNONYM_OF flight attendant
```

オプションの *number* はユーザー定義の関係を識別します。この番号は、シソーラス定義ファイル全体から取られた重複しない数 (現在は 1 ~ 128) でなければなりません。例：RELATED_TO(42)。

アプリケーションで関係名および番号の代わりにシソーラス関係のシンボル名を使用したい場合は、アプリケーションで名前と番号のマッピングを処理する必要があります。例：関係 *opposite_of* を RELATED_TO(1) として定義する場合は、アプリケーションでこの名前を内部関係名 RELATED_TO(1) にマップする必要があります。

associated-term

各関連用語は、関係名の前に置く必要があります。関連用語は、指定した関係について各メンバー用語と関連しています。すべてのメンバー用語がお互いに関連している場合は、メンバー関係を使用してそれを指定できます。

- 最大長は 64 バイトです (コード・ページ UTF-8 の場合は 42 バイト)。
- 同一文字の 1 バイト文字と 2 バイト文字は、同じと見なされます。
- 英大文字および小文字は区別されません。
- 用語にはブランク文字を含めることができます。
- 単一バイト文字のピリオド "." またはコロン ":" は使用できません。

以下に関連用語の例を示します。

```
:WORDS:SYNONYM
  reject
  decline
  RELATED_TO(1) accept
```

第 3 部 付録

付録 A. 移行

DB2 Net Search Extender バージョン 8.1.x は、DB2 テキスト情報エクステンダー バージョン 7.2 の検索インターフェースおよび関数を含むように、広範囲に変更されました。したがって、次の 3 つの移行オプションがあります。

- テキスト情報エクステンダー バージョン 7.2 から Net Search Extender バージョン 8.1.x への移動。

DB2 バージョン 7.2 およびテキスト情報エクステンダー バージョン 7.2 から移行する場合は、DB2 インスタンスおよびインスタンス・データベースの移行から始めます。DB2 インスタンスをバージョン 7 からバージョン 8 に移行するために使用する db2imigr やインスタンス・データベースの移行に使用する db2migrate db <db name> などの DB2 コマンドの正しい構文と使用方法については、適切な DB2 文書を参照してください。

- Net Search Extender バージョン 7.2 から Net Search Extender バージョン 8.1.x への移動
- Net Search Extender バージョン 8.1 から Net Search Extender バージョン 8.1.x および 8.2 への移動

DB2 Net Search Extender の移行プロシージャは db2extmdb と呼ばれ、メッセージを英語で戻します。移行中にエラーが起きた場合は、エラーの修復を試み、もう一度 db2extmdb を呼び出してください。

db2extmdb は移行に関係するすべての db2ext 管理情報を DB2EXT.TMIGRATION という新しい表に収集します。この表では各テキスト索引が 1 行で表されます。この移行情報表は、データベースの移行が正常に完了するまで存在します。この表をドロップすることはできません。

テキスト索引の移行時の処理は、ログ表の状態および索引ファイルの可用性によって異なります。

- ログ表が空であり、索引ファイルにアクセスできる場合は、索引を移行します。これが最も速いオプションです。
- ログ表が空ではないか、あるいは索引ファイルが検出できない場合は、データベース状態の一貫性を保証できないため、索引を再作成します。

移行にかなりの時間がかかる場合のあることにご注意ください。移行を実行中には、ユーザー表に何も変更を加えないでください。

注

最新の移行情報を入手するには、CD-ROM および DB2 Net Search Extender Web サイト上の release.txt ファイルをチェックしてください。

Net Search Extender バージョン 8.1 から Net Search Extender バージョン 8.1.x および 8.2 への移動

Net Search Extender バージョン 8.1 から Net Search Extender バージョン 8.1.x へ移動するには、必要な事前のステップはありません。したがって、古いインストール・システムを除去してから、最新バージョンの Net Search Extender をインストールすることができます。

UNIX と Windows の両方で、db2iupdt を使用して、DB2 インスタンスをバージョン 8.1 からバージョン 8.1.x または 8.2 に移行します。その後、新規の db2extmdb ツールを使用して、このインスタンスの使用可能なすべてのデータベースを移行します。呼び出し元はインスタンス所有者でなければならず、構文は次のようにします。

```
db2extmdb <database name>
```

移行の実行中には、テキスト索引を持つユーザー表に何の変更も加えないようにします。

注意および推奨事項

移行のステップは次のファイルにログとして記録されます。

```
<os-dependent>/sql1lib/db2ext/db2extm <database-name>.log
```

db2extmdb プログラムを呼び出す前に、ユーザーはすべての索引ディレクトリとデータベースのバックアップを取っておく必要があります。

Net Search Extender バージョン 7.2 から Net Search Extender バージョン 8.1.x への移動

Net Search Extender バージョン 7.2 を使用中で、すべての新規機能が不要であれば、CD-ROM からまたはダウンロードによって入手可能な Net Search Extender バージョン 7.2 互換インターフェースをインストールして、古いインターフェースを使用し続けることができます。詳細については、release.txt を参照してください。Net Search Extender の新規機能を使用するには、変更された管理構文に対応するように管理スクリプトを変更し、検索構文を新しい検索インターフェースに変更する必要があります。

Net Search Extender バージョン 7.2 から Net Search Extender バージョン 8.1.x へ自動的に移行する方法はありません。

これらの互換インターフェースは使用すべきでなく、将来のリリースでは使用できなくなることに注意してください。

テキスト情報エクステンダー バージョン 7.2 から Net Search Extender バージョン 8.1.x への移動

UNIX と Windows の両方で、まず DB2 ツール `db2imgr` を使用し、次に `db2iupdt` を使用して、DB2 インスタンスをバージョン 7.2 からバージョン 8.1.x に移行します。例：

```
db2imgr -u <fenced userid> <instance name>
db2iupdt -u <fenced userid> <instance name>
```

その後、`migrate` コマンドと新規の `db2extmdb` ツールを使用して、このインスタンスのすべてのデータベースを移行します。例：

```
db2 migrate database <database name>
db2extmdb <database name>
```

呼び出し元はインスタンス所有者でなければなりません。

プログラム `db2extmdb` は移行に関係のある `db2ext` 管理情報すべてを集めて `DB2EXT.TMIGRATION` という名前の新しい表に移動します。この表では、各テキスト索引が 1 行で表わされます。この移行情報表はデータベースの移行が正常に完了するまで残っていますが、ユーザーがドロップしてはなりません。

エラーが起きた場合は、エラーを修正してもう一度 `db2extmdb` を呼び出します。

テキスト索引の移行中には、別の処理が行われます。この処理の内容は「ログ表」の状態によって異なります。ログ表が空であれば、索引が移行されて、最も早い方法で処理が行われます。ログ表が空ではない場合、データベースとの整合状態が保証されないため、索引を再作成することが必要になります。この処理にはかなり時間がかかる場合があります。

移行の実行中には、テキスト索引を持つユーザー表に何の変更も加えないようにします。

推奨事項

`db2extmdb` プログラムを呼び出す前に、ユーザーはすべての索引ディレクトリとデータベースのバックアップを取っておく必要があります。また、バージョン 7.2 テキスト索引の作成のために使用したすべての文書モデル・ファイルが存在しており、それらを読み取って使用できることを確認しておいてください。

DB2 インスタンスの移行後には、以前の `sqllib` ディレクトリが `sqllib_v71` という名前に変更されます。デフォルトの索引ディレクトリに保管しているテキスト索引がある場合は、`sqllib_v71/db2ext/indexes` ディレクトリを新規の `sqllib` ディレクトリ `sqllib/db2ext/indexes` に移動します。

注意および推奨事項

移行のステップは次のファイルにログとして記録されます。

```
<os-dependent>/sql1lib/db2ext/db2extm <database-name>.log
```

HP-UX または Linux の場合、これらのプラットフォームではテキスト情報エクステンダーがまだ使用できないため、このステップは不要です。

また、DB2 テキスト情報エクステンダー バージョン 7.2 は、今後使用できないことにも注意してください。

付録 B. 大量のメモリーの使用

ストアード・プロシージャ検索にキャッシュを使用すると、大量のメモリーが必要となります。また、以下のプラットフォームではメモリー所要量が異なります。

- AIX
- Windows
- Solaris オペレーティング環境
- Linux
- HP-UX

AIX (32 ビットおよび 64 ビット)

システムしきい値の構成は以下のとおりです。

- コマンド `ulimit -a` を使用してシステムしきい値を確認します。
- 「unlimited」以外の値がある場合は、以下のステップに従います。
 - `root` でログオンします。
 - ファイル `/etc/security/limits` をバックアップしてから、ファイルを編集してハードしきい値を高くします。
 - 使用する DB2 インスタンス所有者に対して、すべての値を「unlimited」(値 -1) に設定します。

共有メモリーしきい値の構成は以下のとおりです。

- AIX では、共有メモリーしきい値を構成する必要はありません。

スワップ・スペースの構成は以下のとおりです。

- コマンド `lsattr -E -l sys0` を使用して、システム RAM サイズを入手します。
- `lspcs -a` コマンドを使用してスワップ・スペースのサイズを入手します。
- スワップ・スペース・サイズをシステム RAM 総量の 1.5 倍か 2 倍以上に設定するか、`CREATE INDEX` コマンドで指定する `MAXIMUM CACHE SIZE` パラメーターを使用します。SMIT ユーティリティーを使用して、大きい方の数値を選択します。

AIX での最大キャッシュ・サイズの制限が約 1536 MB (1.5 GB = 1610612736 バイト) であることに注意してください。

AIX (32 ビット) の場合、ストアード・プロシージャ検索インターフェースを使用して大容量のキャッシュ表を検索するには、`db2fmp` 実行可能モジュールの `maxdata` 設定を変更することが必要です。この変更を行うには、インスタンス所有者としてログオンして、次のコマンドを発行します。

```
db2stop
cd ~/sql11ib/adm
cp db2fmp db2fmp.org /usr/bin/echo '¥0040¥0¥0¥0' | dd of=db2fmp bs=4 count=1
seek=19 conv=notrunc
```

大量のメモリーの使用

それでもメモリーが足りない場合は、次のコマンドを発行して、ヒープ・ストレージの量をさらに減らします。

```
cp db2fmp db2fmp.org /usr/bin/echo '¥0000 ¥0 ¥0 ¥0' | dd of=db2fmp bs=4 count=1 seek=19 conv=notrunc
```

この処理により、maxdata は減少して 1 (256 MB) になります。

元の設定にもどすには、次のコマンドを発行します。

```
db2stop
cd ~/sql1lib/adm
cp db2fmp.org db2fmp
```

この変更により、最大ヒープ・サイズは 1.2 GB から 0.5 GB に減少します。そのため、この DB2 インスタンスを使用する他のアプリケーションに影響が出る可能性があります。

Windows (32 ビット)

ページング・ファイルのサイズの調整は以下のとおりです。

- Windows 仮想メモリー・ページング・ファイル・サイズをシステム RAM の総量の 1.5 倍か 2 倍以上に設定するか、CREATE INDEX コマンドで指定する MAXIMUM CACHE SIZE パラメーターを使用します。大きい方の数値を選択してください。ページング・ファイル・サイズの変更については、Windows の資料を参照してください。

Windows での最大キャッシュ・サイズの制限が約 1000 MB (1 GB = 1073741824 バイト) であることに注意してください。

Solaris オペレーティング環境 (32 ビットおよび 64 ビット)

システムしきい値の構成は以下のとおりです。

- コマンド ulimit -a を使用してシステムしきい値を確認します。
- 次に、以下のステップに従います。
 - root でログオンします。
 - ファイル /etc/system をバックアップしてから、ファイルを編集してハードしきい値を高くします。
 - 以下の行を追加するか、以下の行が示された最小値以上に設定されていることを確認します。

```
rlim_fd_cur -> Default 64, recommended >= 1024
```

```
rlim_fd_cur_max -> Default 1024, recommended >= 4096
```

共有メモリーしきい値の構成は以下のとおりです。

- コマンド sysdef -i を使用して現行設定値を確認します。
- ファイル /etc/system を編集し、set shmsys:shminfo_shmmax=0xffffffff を使用して共有メモリー・サイズのしきい値を設定します。

場合により、以下のパラメーター値も増やす必要があります。

```
set shmsys:shminfo_shmmni=512
```

set shmsys:shminfo_shmseg=128、次にシステムをリブートします。

スワップ・スペースの構成は以下のとおりです。

- コマンド `/usr/sbin/prtconf` を使用して、システム RAM サイズを入手します。
- `swap -l` コマンドを使用してスワップ・スペース・サイズを入手します。
- スワップ・スペース・サイズをシステム RAM の総量の 1.5 倍か 2 倍以上に設定するか、`CREATE INDEX` コマンドで指定する `MAXIMUM CACHE SIZE` パラメーターを使用してください。大きい方の数値を選択してください。

スワップ・スペースを増やす方法については、Solaris システムの資料を参照してください。

Solaris での最大キャッシュ・サイズの制限が約 2000 MB (2 GB = 2147483647 バイト) であることに注意してください。

Linux

Linux 上で推奨されているカーネル・パラメーターについては、DB2 資料「概説およびインストール」を参照してください。

新規 Linux カーネルおよびディストリビューションの妥当性検査状況は頻繁に更新されます。サポートされる Linux ソフトウェア・レベルの最新情報を入手するには、<http://www.ibm.com/software/data/db2/linux/validate> を参照してください。

現在の共有リソース制限について調べるには、`ipcs -l` を使用してください。システムの制限について検査するには、`ulimit -a` コマンドを使用してください。

HP-UX (32 ビットおよび 64 ビット)

HP-UX 上で推奨されているカーネル・パラメーターについては、DB2 資料「概説およびインストール」を参照してください。

付録 C. Net Search Extender インフォメーション・カタログ

DB2 Net Search Extender は、デフォルト設定、構成、テキスト索引、およびフォーマットに関する重要な情報をカタログ表に保管します。表上にいくつかのビューを照会して、この情報を表示することができます。

次のビューおよび表は、使用中のシステムの現行の構成を反映しています。

- データベース・レベル情報ビュー：
 - db2ext.dbdefaults
 - db2ext.proxyinformation 表
- 索引レベル情報ビュー：
 - db2ext.textindexes
 - db2ext.textindexformats
 - db2ext.indexconfiguration

互換性の観点から、DB2 テキスト情報エクステンダー・ビューは引き続き使用可能です。これらは、db2ext.textcolumns、db2ext.models、および db2ext.formats です。

- テキスト索引の表ビュー：
 - イベント・ビュー
 - ログ表ビュー

データベース・レベル情報のビュー

ビュー db2ext.dbdefaults は、データベースのデフォルト値をすべて表示します。

データベース・レベル上のデフォルト値は、変更することができず、このビューで属性値のペアとして選択可能です。

db2ext.dbdefaults

```
db2 select DEFAULTNAME, DEFAULTVALUE from db2ext.dbdefaults
```

表 9. db2ext.dbdefaults ビュー

属性	デフォルト値	備考
CCSID	データベースの CCSID	文書のデフォルト CCSID。これは、CCSID が CREATE INDEX コマンドに指定されていない場合に適用されます。
FORMAT	TEXT	文書のデフォルト・フォーマット。これは、フォーマットが CREATE INDEX コマンドに指定されていない場合に適用されます。

データベース・レベル情報のビュー

表9. db2ext.dbdefaults ビュー (続き)

属性	デフォルト値	備考
INDEXDIRECTORY	備考欄のパス名を参照	フル・テキスト索引ファイル用のディレクトリ。これは、索引ディレクトリが CREATE INDEX コマンドに指定されていない場合に適用されます。パス名: \$DB2EXT_INSTOWNERHOMEDIR/sql1lib/db2ext/indexes
LANGUAGE	EN_US	文書の言語。
MODELCCSID	データベースの CCSID	文書モデル・ファイルの CCSID。
UPDATECOMMITCOUNT	0	更新中に 1 つのトランザクションで処理された変更の数。
CLEARCOMMITCOUNT	0	CLEAR INDEX コマンドの実行中に 1 つのトランザクションで処理された変更の数。
UPDATEFREQUENCY	なし	新規索引での更新をチェックする場合。
UPDATEMINIMUM	1	更新が実行される前の変更の最小数。
WORKDIRECTORY	備考欄のパス名を参照	索引一時ファイル用のディレクトリ。パス名: <os_dependent>/sql1lib/db2ext/indexes
CACHEDIRECTORY	備考欄のパス名を参照	CREATE INDEX コマンドの PERSISTENT CACHE オプションのデフォルト・ディレクトリ。パス名: <os_dependent>/sql1lib/db2ext/memory
PCTFREE	50	将来の挿入に備えて解放されたままのキャッシュのパーセント。
USERPERSISTENTCACHE	1	永続キャッシュを使用します。
AUTOMATICREORG	1	CREATE INDEX コマンドの REORGANIZE オプション。これは自動再編成を意味します。
TREATNUMBERSASWORDS	0	シーケンスを、隣接した文字である場合でも、別個のワードとして解釈しません。たとえば、デフォルトの 0 は、tea42at5 が 1 つのワードと見なされることを意味します。
INDEXSTOPWORDS	1	索引付け中にストップワードを無視する。
VERSION		NSE V8.1.4 Net Search Extender の現行バージョン番号。

表 9. *db2ext.dbdefaults* ビュー (続き)

属性	デフォルト値	備考
UPDATEDELAY	0	タイム・スタンプを使用したインクリメンタル更新では、ログ項目だけが使用されます。ログ項目が現在のタイム・スタンプより古い場合、これは <code>updatedelay</code> となります。これは、更新コマンドの実行中にユーザー表に対して長期実行されるトランザクションのためだけに使用され、更新が失われてしまうことのないようにします。ログ項目とインクリメンタル更新を使用する場合、キャプチャー表は使用されないことに注意してください。

db2ext.proxyinformation 表

データ・リンクを使用中であり、プロキシ・サーバーを使用してファイルにアクセスしたい場合は、`db2ext.proxyinformation` 表でプロキシ情報を指定する必要があります。

表 10. *db2ext.proxyinformation* ビュー

属性	タイプ	備考
PROXYHOST	VARCHAR(254)	プロキシ・サーバーのホスト名。
PROXYPORT	VARCHAR(6)	プロキシ・サーバーの使用ポート。
PROXYTYPE	VARCHAR(10)	プロキシ・サーバーのタイプ (PROXY または SOCKS)。
PROXYTIMEOUT	INTEGER	タイムアウト (秒)。

最大で 1 行を挿入できます。ポート 123 上にプロキシ・サーバーがある場合は、次の SQL ステートメントを使用して行を挿入できます。

```
db2 insert into db2ext.proxyinformation values ('proxy1', '123', 'PROXY', 10)
```

索引レベル情報のビュー

以下の DB2 Net Search Extender ビューを使用して、索引レベルで情報を照会できます。

- `db2ext.textindexes`
- `db2ext.textindexformats`
- `db2ext.indexconfiguration`
- `<index eventview name schema>.<index eventview name>`

旧バージョンとの互換性の理由から、DB2 テキスト情報エクステンダー・ビュー `db2ext.textcolumns`、`db2ext.formats`、および `db2ext.models` は引き続きサポートされていますが、使用すべきではありません。

索引レベル情報のビュー

db2ext.textcolumns ビューでは、OPERATION、OPERATIONBEGIN、および OPERATIONEND 列がサポートされなくなっていることに注意してください。

db2ext.textindexes ビュー

DB2 Net Search Extender に対して使用可能な各データベースには、db2ext.textindexes ビューが含まれています。このビューには、このデータベースに作成されたテキスト索引の設定値、統計およびデフォルトに関する情報が含まれています。

テキスト索引を作成すると、db2ext.textindexes に新規項目が作成されます。作成したテキスト索引をドロップすると、これらの項目も削除されます。

索引に関する情報を得るためにビューを照会できます。次は、索引スキーマを使用した例です。

```
db2 "select COLNAME from db2ext.textindexes where INDSHEMA='myschema'
and INDNAME='myindex'"
```

ただし、通常の SQL データ操作コマンドを使用してビューを変更したり、カタログ・ビューを明示的に作成またはドロップすることはできないことに注意してください。ビューの詳細な内容については、次の表を参照してください。

レプリケーション・パラメーターがこのビューに組み込まれていないことにも注意してください。

表 11. db2ext.textindexes ビュー

属性	タイプ	備考
INDSCHEMA	VARCHAR(128)	テキスト索引のスキーマ名。
INDNAME	VARCHAR(128)	テキスト索引の名前。
TABSCHEMA	VARCHAR(128)	基本表、ニックネーム、およびビューのスキーマの表名。
TABNAME	VARCHAR(128)	索引が作成された別名。
COLNAME	VARCHAR(128)	索引が作成された列。
CCSID	INTEGER	この索引の文書 CCSID。
LANGUAGE	VARCHAR(5)	この索引の文書の言語。
FUNCTIONSCHEMA	VARCHAR(128)	列マッピング機能のスキーマ。
FUNCTIONNAME	VARCHAR(18)	列マッピング機能の名前。
INDEXDIRECTORY	VARCHAR(256)	フル・テキスト索引ファイル用のディレクトリ。
WORKDIRECTORY	VARCHAR(256)	索引一時ファイル用のディレクトリ。
CACHEDIRECTORY	VARCHAR(256)	永続キャッシュのディレクトリ (persistentcache=1 の場合)。
UPDATEFREQUENCY	VARCHAR(300)	この索引に自動更新を適用するためのトリガー基準。
UPDATEMINIMUM	INTEGER	更新を実行する前に変更が必要な文書の最小数。
EVENTVIEWSHEMA	VARCHAR(128)	この索引用に作成されるイベント・ビューのスキーマ。

表 11. db2ext.textindexes ビュー (続き)

属性	タイプ	備考
EVENTVIEWNAME	VARCHAR(128)	この索引用に作成されるイベント・ビューの名前。
LOGVIEWSCHEMA	VARCHAR(128)	この索引用に作成されるログ・ビューのスキーマ。
LOGVIEWNAME	VARCHAR(128)	索引用に作成されるログ・ビューの名前 (ビュー上のインクリメンタル更新にとって重要)。
COMMITCOUNT	INTEGER	コミット・カウント更新用のデフォルト。
NUMBER_DOCS	INTEGER	索引内の現在の文書数の合計。索引の更新中にこの値が更新されるのは、commitcount が設定されている場合だけであることを注意。
REORG_SUGGESTED	INTEGER	UPDATE INDEX REORGANIZE の実行によってパフォーマンス向上が可能かどうかを示す。このパラメーターが真 (1) になるのは、最低 1 つのノードで索引の再編成が提案されている場合だけである。
REORGAUTOMATIC	INTEGER	更新操作時に索引が自動的に再編成された場合は、1。
RECREATEONUPDATE	INTEGER	更新操作時に索引が自動的に再編成された場合は、1。
CREATIONTIME	TIMESTAMP	索引作成の時刻。
UPDATETIME	TIMESTAMP	最終の更新の時刻。UPDATE TIME が CREATION TIME と等しい場合は、更新は処理されていません。
PERSISTENTCACHE	INTEGER	永続キャッシュが使用される場合は、1。
MAXIMUMCACHESIZE	INTEGER	キャッシュの最大サイズ。
PCTFREE	INTEGER	将来の挿入に備えて解放されたままのキャッシュのパーセント。
CACHETABLE	VARCHAR(32000)	CACHE TABLE 用の列式リスト。
RESULTORDER	VARCHAR(32000)	INITIAL RESULT ORDER 用の SQL による順序付け。
ATTRIBUTES	VARCHAR(32000)	ATTRIBUTES 用の列式リスト。
VIEWKEYCOLUMNS	VARCHAR(32000)	ビュー上の索引のキー列。

db2ext.indexconfiguration ビュー

索引構成パラメーターは、db2ext.indexconfiguration ビューで使用できます。ビューは、通常の SQL 照会機能を介して使用ができます。次は、索引名を使用した例です。

索引レベル情報のビュー

```
db2 "select VALUE from db2ext.indexconfiguration where INDSHEMA='myschema'
and INDNAME='myindex' and PARAMETER ='INDEXSTOPWORDS'"
```

ビューの詳細な内容については、次の表を参照してください。

表 12. db2ext.indexconfiguration ビュー

属性	タイプ	備考
INDSCHEMA	VARCHAR(128)	索引のスキーマ名。
INDNAME	VARCHAR(128)	索引の名前。
PARAMETER	VARCHAR(30)	パラメーターのタイプ。
VALUE	VARCHAR(512)	パラメーターの値。

PARAMETER 属性と VALUE 属性には、選択可能な値がいくつかあります。

表 13. db2ext.indexconfiguration ビュー

属性と値	属性と値
PARAMETER	VALUE
- TREATNUMBERASWORDS	- 0 または 1
- INDEXSTOPWORDS	- 0 または 1
- UPDATEDELAY	- 秒数 >= 0

詳細については、CREATE INDEX コマンドの CONFIGURATION オプションを参照してください。

db2ext.textindexformats ビュー

索引のフォーマットおよびモデル情報は、db2ext.textindexformats ビューで入手できます。次は、索引名を使用した例です。

```
db2 "select FORMAT from db2ext.textindexformats where INDSHEMA='myschema'
and INDNAME='myindex'"
```

ビューの詳細な内容については、次の表を参照してください。

表 14. db2ext.textindexformats ビュー

属性	タイプ	備考
INDSCHEMA	VARCHAR(128)	索引のスキーマ名 (ログ表での索引名およびスキーマ名の接頭部として使用)。
INDNAME	VARCHAR(128)	CREATE INDEX コマンドで指定される索引名。
FORMAT	VARCHAR(30)	モデルはこのフォーマットにバインドされています。
MODELNAME	VARCHAR(30)	文書モデルの名前。
MODELFILE	VARCHAR(256)	モデル定義を含むファイル。
MODELCCSID	INTEGER	MODELFILE の CCSID。
DEFAULT	INTEGER	現行は 1。索引での複数のフォーマットは現在サポートされていないため。

テキスト索引の表ビュー

以下の DB2 Net Search Extender ビューを使用して、索引レベルで情報を照会できます。

- イベント・ビュー
- ログ表ビュー

イベント・ビュー

文書が見つからない、など索引付け中に問題が発生した場合、このビューを使用すれば、索引付けの状況およびエラー・イベントに関する情報を入手できます。これらの索引更新イベントは、その後、索引のイベント表に書き込まれます。

スキーマおよび名前は、db2ext.textindexes ビューに保管されます。イベント・ビューの名前を取得するには、次の例を使用します。

```
db2 "select EVENTVIEWSHEMA, EVENTVIEWNAME from db2ext.textindexes
     where INDSHEMA = 'myschema' and INDNAME = 'myindex'
```

索引のイベント・ビューは、次の列から構成されています。

表 15. イベント・ビュー

属性	タイプ	備考
OPERATION	INTEGER	フル・テキスト索引に反映されるユーザー表上の操作 (挿入 = 0/更新 = 1/削除 = 2)。 レプリケーション・キャプチャー表を使用している場合は、更新操作が削除操作と挿入操作に分けられます。このとき、イベント表への挿入操作は、索引が作成されたソース表での挿入または更新操作のいずれかによるものである場合があります。
TIME	TIMESTAMP	イベント項目作成のタイム・スタンプ。
REASON	INTEGER	理由コード。理由コードのリストについては、249 ページの『付録 I. テキスト検索エンジン理由コード』を参照してください。
SEVERITY	INTEGER	表の項目の重大度。たとえば、1 は通知用、4 は警告、8 は表の項目のエラーを示す。
MESSAGE	VARCHAR(1024)	追加のテキスト情報。
KEY1, ... KEY14	ユーザー表に応じて異なる	ユーザー表の最初の主キー列から最後の主キー列 (最大 14)。
PARTITION	INTEGER	このエラーが発生したデータベース・パーティション番号。分散環境でない場合は 0。

イベントは、DB2TEXT CLEAR EVENTS コマンドを使用してクリアできます。詳細については 118 ページの『CLEAR EVENTS』を参照してください。

注

更新処理の開始、コミット、および終了などの情報イベントは、このビューでも選択可能です。

この場合、Key1、... Key14 および OPERATION は、すべて NULL 値が指定されています。

ビュー上の索引の場合、PK01、...、PK14 列は、CREATE INDEX コマンドの KEY COLUMNS 文節で指定された列に関係しています。

ログ表、ビュー、およびニックネーム

ログ表の目的は、外部のフル・テキスト索引との同期を必要とするユーザー表またはビュー上に変更操作を保管することにあります。

通常の表またはニックネーム表上で作成された索引の場合、変更情報をログ表に送るためにユーザー表上に作成されたトリガーがあります。ただし、レプリケーション・キャプチャー表を使用している場合、ログ表は作成されず、代わりにレプリケーション・キャプチャー表が使用されます。

ログ表の場合、更新コマンドは項目を読み取り、正常な同期の後、削除します。

ただし、ビューに関する索引の場合、トリガーはログ表に書きこむことができません。ビューは更新できるので、この作業はユーザーが行います。

表 16. ログ表ビュー

属性	タイプ	備考
OPERATION	INTEGER	索引の同期が必要となるユーザー表の変更タイプ: (0 = 挿入、1 = 更新、2 = 削除)。
TIME	TIMESTAMP	この表の行の作成時のタイム・スタンプ。
PK01 ... PKnm	ユーザー表と同じ	エラーの場合は、問題が起きた列。これらは、ビュー上の索引の場合のユーザー表の主キー列または同等のキー列のコピーです。

表を作成したユーザーは、このビューを更新、挿入、および削除できます。

索引の作成コマンドにレプリケーション・キャプチャー表を指定する場合、ログ表は作成されず、代わりにレプリケーション・キャプチャー表が使用されます。レプリケーション・キャプチャー表には、以下の列が含まれていることが必要です。

表 17. レプリケーション・キャプチャー表

属性	タイプ	備考
IBMSNAP_OPERATION	INTEGER	索引の同期が必要となる CD または CCD 表の変更タイプ: (I = 挿入、U = 更新、D = 削除)。 レプリケーション・キャプチャー表を使用している場合は、更新操作が削除操作と挿入操作に分けられます。このとき、イベント表への挿入操作は、索引が作成されたソース表での挿入または更新操作のいずれかによるものである場合があります。
IBMSNAP_COMMITSEQ	CHAR	CD または CCD 表の対応する列へのマップ。
IBMSNAP_INTENTSEQ	CHAR	CD または CCD 表の対応する列へのマップ。
PK01 ... PKnm	ユーザー表と同じ	エラーの場合は、問題が起きた列。これらの列はユーザー表の主キー列です。

表を定義したユーザーは、 GRANT オプションを指定して更新、挿入、および削除できます。

付録 D. サポートされている CCSID

次の CCSID は、DB2 Net Search Extender でサポートされています。

CCSID

37	米国/カナダ - CECP
273	ドイツ F.R./オーストリア - CECP
274	旧ベルギー・コード・ページ
277	デンマーク、ノルウェー - CECP
278	フィンランド、スウェーデン - CECP
280	イタリア - CECP
284	スペイン/ラテンアメリカ - CECP
285	英国 - CECP
290	日本語 (カタカナ) 拡張
297	フランス - CECP
301	日本 DB PC
367	ASCII
420	アラビア語バイリンガル
423	ギリシャ - 183
424	イスラエル (ヘブライ語)
437	米国英語
500	国際 #5
737	MS DOS ギリシャ語
806	ヒンディ語
813	ギリシャ語
819	Latin-1
833	韓国語拡張
836	中国語 (簡体字) 拡張
838	タイ語 (低声調記号および古代文字付)
848	PC、キリル文字、ウクライナ語 (ユーロ文字付)
850	Latin-1
852	Latin-2
855	ブルガリア語
857	トルコ語

858	パーソナル・コンピューター - マルチリンガル (ユーロ文字付)
860	ポルトガル語
862	ヘブライ語
863	カナダ英語
864	アラビア語
866	ロシア語
867	イスラエル - パーソナル・コンピューター
869	ギリシャ語
870	Latin 2 - EBCDIC マルチリンガル
871	アイスランド - CECP
872	キリル文字 - PC (ユーロ文字付)
874	タイ語
875	ギリシャ
891	韓国 - パーソナル・コンピューター
895	日本 (7 ビット・ローマ字)
901	PC バルト語、マルチリンガル (ユーロ文字付)
902	8 ビット・エストニア語 (ユーロ文字付)
912	Latin-2
915	ロシア語
916	ヘブライ語
920	トルコ語
921	ラトビア語、エストニア語
922	エストニア語
923	Latin 9
924	Latin 9 EBCDIC
927	台湾 PC
930	日本 EBCDIC
932	日本語、結合された SBCS/DBCS
933	韓国語
935	中国語 (簡体字)
937	中国語 (繁体字)
938	台湾 PC
939	日本語
941	日本 OPEN
942	日本語、結合された SBCS/DBCS
943	日本語、結合された SBCS/DBCS

948	中国語 (繁体字)、結合された SBCS/DBCS
949	韓国語
950	中国語 (繁体字)、結合された SBCS/DBCS
954	日本語
964	中国語 (繁体字)、結合された SBCS/DBCS
970	韓国語
1025	キリル文字、マルチリンガル
1026	Latin #5 - トルコ
1027	日本語 (ローマ字) 拡張
1040	韓国語拡張 - パーソナル・コンピューター
1041	日本語拡張 - パーソナル・コンピューター
1043	中国語 (繁体字) 拡張 - PC
1046	アラビア語
1047	Latin 1/Open Systems
1051	H-P エミュレーション、Roman 8
1088	改訂韓国語 - パーソナル・コンピューター
1089	アラビア語
1112	バルト語 - マルチリンガル、EBCDIC
1115	中華人民共和国 (PRC)-PC
1122	エストニア、EBCDIC
1123	キリル文字、ウクライナ
1124	ウクライナ語
1125	ウクライナ語
1131	ベトナム語
1137	デーバナーガリー語 EBCDIC
1140	米国、カナダ、その他 ECECP
1141	オーストリア、ドイツ ECECP
1142	デンマーク、ノルウェー ECECP
1143	フィンランド、スウェーデン ECECP
1144	イタリア ECECP
1145	スペイン、ラテンアメリカ (スペイン語) ECECP
1146	英国 ECECP
1147	フランス ECECP
1148	国際 ECECP
1149	アイスランド ECECP
1153	EBCDIC Latin 2 マルチリンガル (ユーロ文字付)

1154	EBCDIC キリル文字 マルチリンガル (ユーロ文字付)
1155	EBCDIC トルコ語 (ユーロ文字付)
1156	EBCDIC バルト語マルチ (ユーロ文字付)
1157	EBCDIC エストニア語 (ユーロ文字付)
1158	EBCDIC キリル文字、ウクライナ語 (ユーロ文字付)
1160	タイ語 (低声調記号および古代文字付)
1161	タイ語 (低声調記号および古代文字付) PC
1162	タイ語 MS Windows
1163	ベトナム語 ISO-8 (ユーロ文字付)
1164	ベトナム語 EBCDIC (ユーロ文字付)
1200	UCS2
1208	UTF8
1250	Latin-2、ベロルシア語
1251	ロシア語
1252	Latin-1
1253	チェコ語
1254	トルコ語
1255	ヘブライ語
1256	アラビア語
1258	ベトナム語
1275	Apple、Latin 1
1280	Apple ギリシャ語
1281	Apple トルコ語
1282	Apple 中央ヨーロッパ
1283	Apple キリル文字
1351	日本 OPEN
1363	韓国語
1364	韓国語
1381	中国語 (簡体字)、結合された SBCS/DBCS
1383	中国語 (簡体字)、結合された SBCS/DBCS
1386	中国語 (簡体字)、結合された SBCS/DBCS
1388	中国語 (簡体字)、結合された SBCS/DBCS
1390	日本語
1392	中国 GB18030
1394	Shift JIS X0213
1399	日本 EBCDIC

4909	ギリシャ/Latin ASCII
4930	韓国 DB EBCDIC
4933	中国 EBCDIC
4971	ギリシャ EBCDIC
5026	日本語カタカナ
5035	日本語英数小文字
5039	日本語、結合された SBCS/DBCS
5210	中国 SB PC
5346	Windows Latin-2
5347	Windows キリル文字
5348	Windows Latin-1
5349	Windows ギリシャ
5350	Windows トルコ
5351	Windows ヘブライ語 (ユーロ文字付)
5352	Windows アラビア語
5353	Windows バルト語
5354	ベトナム語
9044	Latin-2 PC
9048	ヘブライ語 PC
9049	トルコ PC
9061	ギリシャ PC
9238	アラビア語 - PC
12712	ヘブライ語 EBCDIC
13121	韓国 SB EBCDIC
13488	UCS2
17248	アラビア語 PC
17584	UCS-2
18030	中国語文字標準
21427	台湾 BIG-5
33722	IBMeucJP
61955	

付録 E. サポートされている言語

DB2 Net Search Extender で指定できる言語パラメーターは、次のとおりです。 Net Search Extender では、言語独自の有効な処理が行われるのは、タイ語で書かれた文書を処理する場合か、ストップワードを処理する場合 (要求された場合) だけです。詳細については、245 ページの『ストップワード』を参照してください。

AR_AA	アラビア語/アラビア語圏
BE_BY	ベラルーシ語/ベラルーシ
BG_BG	ブルガリア語/ブルガリア
CA_ES	カタロニア語/スペイン
CS_CZ	チェコ語/チェコ
DA_DK	デンマーク語/デンマーク
DE_CH	ドイツ語/スイス
DE_DE	ドイツ語/ドイツ
EL_GR	ギリシャ語/ギリシャ
EN_AU	英語/オーストラリア
EN_BE	英語/ベルギー
EN_GB	英語/英国
EN_US	英語/米国
EN_ZA	英語/南アフリカ
ES_ES	スペイン語/スペイン
ET_EE	エストニア語/エストニア
FI_FI	フィンランド語/フィンランド
FR_BE	フランス語/ベルギー
FR_CA	フランス語/カナダ
FR_CH	フランス語/スイス
FR_FR	フランス語/フランス
HE_IL	ヘブライ語/イスラエル
HI_IN	ヒンディ語/インド
HR_HR	クロアチア語/クロアチア
HU_HU	ハンガリー語/ハンガリー
ID_ID	インドネシア語/インドネシア
IT_CH	イタリア語/スイス
IW_IL	ヘブライ語/イスラエル
IT_IT	イタリア語/イタリア

言語

JA_JP	日本語/日本
KO_KR	韓国語/韓国
LT_LT	リトアニア語/リトアニア
LV_LV	ラトビア語/ラトビア
MK_MK	マケドニア語/マケドニア
MS_MY	マレー語/マレーシア
NB_NO	ノルウェー語ブークモール/ノルウェー語
NL_BE	オランダ語/ベルギー
NL_NL	オランダ語/オランダ
NN_NO	ノルウェー語ニーノシュク/ノルウェー語
NO_NO	ノルウェー語/ノルウェー
PT_BR	ポルトガル語/ブラジル
PL_PL	ポーランド語/ポーランド
PT_PT	ポルトガル語/ポルトガル
RO_RO	ルーマニア語/ルーマニア
RU_RU	ロシア語/ロシア
SH_SP	セルビア語 (ローマ字)/セルビア
SK_SK	スロバキア語/スロバキア
SL_SI	スロベニア語/スロベニア
SQ_AL	アルバニア語/アルバニア
SR_SP	セルビア語 (キリル文字)/セルビア
SV_SE	スウェーデン語/スウェーデン
TA_IN	タミール語/インド
TE_IN	テルグ語/インド
TH_TH	タイ語/タイ
TR_TR	トルコ語/トルコ
UK_UA	ウクライナ語/ウクライナ
VI_VN	ベトナム語/ベトナム
ZH_CN	中国語/中国
ZH_TW	中国語/台湾

付録 F. Net Search Extender メッセージ

DB2 Net Search Extender は、以下のメッセージ・タイプを提供します。

- 通知および警告メッセージ
- エラー・メッセージ

検索関数から戻される SQL 状態コードは、38600 に CTE エラー番号を加えたものです。

通知および警告メッセージ

CTE0001	操作は正常に完了しました。
CTE0002	更新およびロックング・サービスは稼働中です。
CTE0003	索引の更新が開始されました。
CTE0004	索引の更新が終了しました。
CTE0005	索引更新のコミット: "%1"、"%2"、"%3" の文書の挿入、更新、または削除、あるいはこれらすべての操作が正常に行われました。
CTE0006	テキスト索引のアクセスに問題が発生しました。詳細に関しては db2diag.log をチェックしてください。
CTE0007	セクション "%1" がどの文書でも検出されませんでした。あるいは無効な文書モデル・セクション名です。
CTE0008	索引の再編成が開始されました。
CTE0009	索引の再編成が終了しました。
CTE0010	属性 "%1" は無効です。
CTE0011	キャッシュのアクティブ化が開始されました。
CTE0012	キャッシュのアクティブ化が終了しました。
CTE0013	永続キャッシュが除去されました。
CTE0014	キャッシュが使用不能になりました。

エラー・メッセージ

CTE0100 DB2 操作が失敗しました。DB2 情報:
"%2" "%4"。

説明: DB2 エラーが発生し、これ以上の処理はできません。

処置: この DB2 エラーの詳細に関しては、db2 ?
SQLxxx を使用してください。

CTE0101 検索エンジン操作が失敗しました。理由コ
ード: "%2"、"%3"、"%4"、
"%5"、"%6"。

説明: 検索エンジン・エラーが発生し、これ以上の処理
はできません。

処置: 詳細については、検索エンジン理由コードの説明
を参照してください。

CTE0102 一般システム機能が失敗しました。エラー
: "%2"。

説明: システム・エラーが発生し、これ以上の処理はで
きません。

処置: 追加情報に関しては、errno.h ヘッダー・ファイ
ルの UNIX を参照してください。

CTE0103 内部エラーが発生しました。ロケーション
: "%1"、"%2"。

説明: 内部処理エラーが発生し、これ以上の処理はでき
ません。更新サービスとロッキング・サービス、および
DB2 の開始および停止を試みてください。

処置: エラーが継続する場合は、トレースを開始して、
db2diag.log もチェックしてください。

CTE0104 メモリー割り振りエラー (検索エンジン)
です。

説明: システムでメモリー不足が発生しました。

処置: インスタンス所有者が使用できるメモリー・サイ
ズを増やすか、並列して実行中の他の処理を停止してく
ださい。

CTE0105 メモリー割り振りエラーです。

説明: システムでメモリー不足が発生しました。

処置: ユーザーが使用できるメモリー・サイズを増やす
か、並列して実行中の他の処理を停止してください。

CTE0106 表 "%1"."%2" には主キーがありません。

説明: 主キーを持たない表に索引を作成しようとしてしま
した。

処置: db2 alter table を呼び出して、主キーがあるか
確認してください。そして、再度、索引を作成してくだ
さい。

CTE0107 ディレクトリー "%1" が存在しません。

説明: 指定したディレクトリーは、存在しません。

処置: ディレクトリーを作成して、インスタンス所有者
がアクセスできることを確認してください。その後、再
度、ディレクトリーを指定してください。分散 DB2 環
境では、このディレクトリーはすべての物理ノード上に
存在する必要があることに注意してください。

CTE0108 オブジェクト "%1"."%2" のキー列の内部
サイズ "%4" が、可能な最大サイズ
"%3" よりも大きなサイズです。

説明: キー列の内部表記が最大サイズを超えています。

処置: 表のレイアウトを変更してから、再度、索引を作
成してください。より小さなキー列を使用してくださ
い。パフォーマンスも向上します。

CTE0109 オブジェクト "%1"."%2" のキー列の数
"%3" が、可能な最大サイズ "%4" より
も大きなサイズです。

説明: キー列の最大数は 14 です。

処置: 表のレイアウトを変更してから、再度、索引を作
成してください。

CTE0111 ファイル "%1" が読み取り可能ではありません。

説明: 指定したファイルが読み取れません。

処置: ファイルに対するアクセス権限をチェックしてく
ださい。ストアード・プロシージャは、このファイル
上で作業する権限も必要である fenced ユーザー ID と
して実行されることを考慮してください。

CTE0112 ファイル "%1" をオープンできません。

説明: 指定したファイルがオープンできません。

処置: ファイルが正しく指定されていることを確認して
ください。

CTE0113 UTF8 エンコードへのモデル・ファイル "%1" の変換エラー。

説明: 指定した CCSID またはデフォルトのデータベース CCSID がモデル・ファイルの CCSID と一致しません。

処置: モデル・ファイルの CCSID の指定が正しいことを確認してください。

CTE0114 ファイル "%2" の文書モデル "%1" を登録できません。

説明: モデル・ファイルが使用できませんでした。

処置: モデル・ファイルの構文が正しく指定されていることを確認してください。

CTE0115 ロッキングの問題が発生しました。ロック・マネージャー情報: "%1" "%2"。

説明: 内部ロッキングの問題が発生しました。

処置: db2text control コマンドを使用して、現行のロックを確認します。同じコマンドを使用して、ペンディング・ロックをクリーンアップしてください。問題が解決しない場合は、ロッキング・サービスおよび更新サービスを停止して再始動してください。

CTE0116 操作が既存のロックと矛盾します。

説明: 他のコマンドがこの索引に関して実行されているときに、現在、許可されないコマンドを使用しようとしてしました。

処置: この索引上に保留されているロックを確認して、どのコマンドが現在、実行中であるか調べてください。他のコマンドが完了するまで待ってください。操作が完了した後も引き続きロックがアクティブの場合は、この索引のロックをクリーンアップして再試行してください。

CTE0117 データベースに使用可能なロック・スペースがすべて使用されます。構成を変更してください。

説明: ロック・ファイルで構成されているデータベース以上のデータベースで作業を試みました。

処置: ロック構成 db2ext1m.cfg の、並列して作業したいデータベースの数を変更してください。コマンド db2text stop および db2text start を使用して、更新サービスおよびロッキング・サービスを再始動してください。

CTE0118 データベースの索引に使用可能なロック・スペースがすべて使用されています。構成を変更してください。

説明: あるデータベースで、ロック・ファイルで構成されている索引より多くの索引に関して作業を試みしました。

処置: ロック構成ファイル db2ext1m.cfg の、並列して作業したい索引の数を変更してください。コマンド db2text stop および db2text start を使用して、更新サービスおよびロッキング・サービスを再始動してください。

CTE0119 索引のロックに使用可能なスペースがすべて使用されています。

説明: 実行中の操作は、ロック構成ファイルで構成されているロック以上のロックがある索引に対して必要です。

処置: ロック構成 db2ext1m.cfg の、並列して作業したいロックの数を変更してください。コマンド db2text stop および db2text start を使用して、更新サービスおよびロッキング・サービスを再始動してください。

CTE0120 更新およびロッキング・サービス構成ファイル・エラーです。

説明: 構成ファイル db2ext1m.cfg にエラーがあります。

処置: db2ext1m.cfg ファイルをチェックし、エラーを訂正してください。コマンド db2text start を使用して、更新サービスおよびロッキング・サービスを再始動してください。

CTE0121 更新およびロッキング・サービス構成ファイルを開くことができません。

説明: ファイル db2ext1m.cfg を開くことができませんでした。

処置: ファイルが存在するかどうか、アクセス可能であるかどうかを確認してください。ファイルにアクセスできない場合は、db2iupdt を使用して db2 インスタンスを更新してください。

CTE0122 更新およびロッキング・サービス構成ファイルで構文エラーが検出されました。

説明: 更新およびロッキング・サービス構成ファイルで構文エラーが検出されました。

処置: 更新およびロッキング・サービス構成ファイルにエラーがないか、チェックしてください。

CTE0126 更新およびロックング・サービス、入力ファイル "%1" が壊れています。

説明: 更新サービスおよびロックング・サービスに必要なファイルが破壊されています。

処置: ファイルが存在するかどうか、アクセスできるかどうかを確認してください。ファイルにアクセスできる場合は、ファイルを名前変更して更新およびロックング・サービスを再始動してください。もう一度ファイルを作成する必要があります。ただし、この処置を行うと、索引作成のために指定した頻度更新はすべて除去されます。

CTE0127 更新およびロックング・サービス構成ファイル・エラーが発生しました。理由コード: "%1"。

説明: 更新およびロックング・サービス領域で内部エラーが発生しました。

処置: DB2 および Net Search Extender を停止して、共有リソースをクリーンアップし、両方を再び始動してください。問題が解決しない場合は、問題を IBM 担当員にお知らせください。

CTE0129 NULL 値をパラメーターとして渡すことができません。

説明: DB2 は、内部ユーザー定義関数に NULL 値を渡しました。

処置: 最初に、指定した基本表が主キーをもっていることを確認してください。この問題を回避するために SELECT ステートメントを変更してください。トレース機能をオンに切り替え、得られた情報を IBM サービス担当者にお渡しください。

CTE0130 指定された検索引き数が最大長を超えています。現在の検索引き数の長さは "%1" で、サポートされる最大長は "%2" です。

説明: 指定した検索引き数の長さが "%1" です。最大長は、"%2" を超えてはなりません。

処置: 検索引き数の長さを "%2" に短縮してください。

CTE0131 ユーザー定義関数 "%1"."%2" が存在しません。

説明: 指定したユーザー定義関数がこのデータベースに存在しません。

処置: このユーザー定義関数の指定した名前を確認する

か、使用中のデータベースにユーザー定義関数を登録してください。

CTE0132 テキスト索引 "%1"."%2" が存在しません。

説明: 指定したテキスト索引がこのデータベースに存在しません。

処置: 指定した名前および使用しているデータベースを確認してください。既存のテキスト索引を確認するには、db2ext.textcolumns ビューを使用します。

CTE0133 テキスト索引 "%1"."%2" はすでに存在します。

説明: 指定したテキスト索引は、すでにこのデータベースに存在します。

処置: 指定した名前および使用しているデータベースを確認してください。既存のテキスト索引を確認するには、db2ext.textcolumns ビューを使用します。

CTE0135 オブジェクト "%1"."%2" が存在しません。

説明: 指定したオブジェクト名がこのデータベースに存在しません。

処置: 指定したオブジェクト名および使用しているデータベースを確認してください。

CTE0136 "%2"."%3" に列 "%1" が存在しません。

説明: 指定した列が存在しません。

処置: 指定した列名を確認してください。使用している表、ビュー、またはデータベースを確認してください。

CTE0137 表スペース "%1" が存在しません。

説明: 指定した表スペースがこのデータベースに存在しません。

処置: 指定した名前および使用しているデータベースを確認してください。

CTE0138 表スペース "%1" が REGULAR ではありません。

説明: 指定した表スペースが REGULAR ではありません。イベント表は、REGULAR 表スペースにのみ作成できます。

処置: REGULAR 表スペースを指定して再びこのコマンドを使用してください。

CTE0139 環境変数 "%1" が設定されていません。

説明: 必要な環境変数が設定されていません。

処置: 環境を確認し、必要な変数を指定して、再びコマンドを使用してください。

CTE0140 データベース "%1" がすでにテキストで使用可能になっています。

説明: 指定したデータベースは、テキストに対してすでに使用可能になっています。

処置: 指定した名前を確認してください。また暗黙接続を暗黙指定する DB2DBDFT 変数を確認してください。

CTE0141 データベース "%1" がテキストで使用可能になっていません。

説明: 指定したデータベースは、テキストに対して使用可能になっていません。

処置: 指定したデータベース名、および DB2DBDFT 変数を確認してください。データベース名が正しい場合は、コマンド db2text enable database for text を使用してください。

CTE0142 コマンドが、ユーザー "%3" に付与された "%1"."%2" に対するコントロール権限を必要とします。

説明: ユーザーが、このコマンドを使用する権限を持っていません。

処置: この表の所有者のみがこのコマンドを使用できるか、必要な権限を付与できます。

CTE0143 コマンドが、ユーザー "%1" のデータベース管理者権限を必要とします。

説明: ユーザーが、このコマンドを使用する権限を持っていません。

処置: データベースの所有者のみがこのコマンドを使用できるか、必要な権限を付与できます。

CTE0144 データベース "%1" に、少なくとも 1 つのアクティブなテキスト索引があります。

説明: すべてのテキスト索引がドロップされるまでは、データベースを使用不可にすることはできません。

処置: 既存の索引に関しては、db2ext.textcolumns ビューを参照してください。DROP INDEX コマンドを使用して、既存の索引をドロップするか、DISABLE DATABASE コマンドで FORCE オプションを指定してください。

CTE0145 CCSID "%1" がサポートされていません。

説明: 指定した CCSID はサポートされていません。

処置: 有効な CCSID を指定してください。

CTE0146 言語 "%1" がサポートされていません。

説明: 指定した言語はサポートされていません。

処置: 有効な言語を指定してください。

CTE0147 フォーマット "%1" がサポートされていません。

説明: 指定したフォーマットはサポートされていません。

処置: 有効なフォーマットを指定してください。

CTE0148 指定されたフォーマット "%1" ではモデル・ファイルを使用できません。

説明: フォーマット "%1" は、モデル・ファイルをサポートしていません。

処置: モデル・ファイルを受け入れるフォーマットを使用するか、モデル・ファイルをコマンドから除去してください。

CTE0149 索引更新頻度に指定された条件 (先頭が "%1") が多すぎます。

説明: 更新頻度の構文が正しくありません。

処置: DAY、HOUR、および MINUTE のパラメーターを一度しか指定しないようにしてください。

CTE0150 予期しないコマンドの終わりです。コマンド構文を確認してください。

説明: コマンドの構文が正しくありません。

処置: コマンド構文を確認してください。必要パラメーターが指定されていることを確認してください。

CTE0151 トークン "%1" は予期されていません。コマンド構文を確認してください。

説明: コマンドの構文が正しくありません。

処置: コマンド構文をチェックして、使用中のトークンが特定のコマンドで許可されていることを確認してください。

エラー・メッセージ

CTE0152 トークン "%1" が長すぎます。

説明: トークンが長過ぎます。

処置: コマンド構文をチェックして、トークンが許容される最大サイズにまで短縮されていることを確認してください。

CTE0153 更新頻度にトークン "%1" が 2 つありません。

説明: 更新頻度に対して誤った構文が指定されました。

処置: DAY、HOUR、および MINUTE のパラメーターを一度しか指定しないようにしてください。

CTE0154 "%2" の値 "%1" は範囲外です。有効な範囲は "%3" ~ "%4" です。

説明: 誤った値が指定されました。値は、許容範囲内であればなりません。

処置: コマンドを更新してください。値を変更して許容範囲内の値になるようにしてください。

CTE0155 検索ストリングが空です。

説明: 空の検索ストリングが指定されました。

処置: 検索ストリングに有効な英数字が含まれているかチェックしてください。

CTE0157 "%1" の前後に構文エラーがあります。

説明: 誤った検索構文が指定されました。

処置: %1 付近の構文を確認してください。訂正してから、再び試してください。

CTE0158 フリー・テキスト検索ストリングがありません。

説明: フリー・テキスト・ストリングを指定してください。

処置: "is about" の後の検索ストリングに有効な英数字が含まれているかチェックしてください。

CTE0159 検索ストリングが許可される長さ "%1" を超えています。

説明: 検索ストリングが長過ぎます。

処置: 検索ストリングのサイズを短縮し、操作を再試行してください。

CTE0160 検索ストリングにセクション名が指定されていません。

説明: 有効なセクション名を指定する必要があります。

処置: 有効なセクション名を追加し、操作を再試行してください。

CTE0162 `escape` コマンドを処理できませんでした。

説明: 検索ストリングに含まれている、マスク文字として使用できる特殊文字が多すぎます。

処置: 検索項目の特殊文字の数を減らすか、エスケープ・コマンドの使用を避けてください。次の特殊文字が使用できます。! * + , _ . : ; { } ~ | ? [] ` = ¥

CTE0163 シソーラス文節にシソーラス名が指定されていません。

説明: シソーラス検索がシソーラス名を指定せずに要求されました。

処置: シソーラス名を検索引き数に指定してください。

CTE0164 シソーラス関係 "%1" に構文エラーがあります。

説明: 指定したシソーラス関係の構文が正しくありません。

処置: 構文仕様に従ってシソーラス関係を書き換えてください。

CTE0166 検索照会では、フリー・テキストは最後のステートメントでなければなりません。

説明: "is about" トークンの後にさらに演算子を指定することはできません。

処置: 照会ストリングを書き換えてください。最後の演算子を "is about" にする必要があります。

CTE0167 フリー・テキスト照会 "%1" に構文エラーがあります。

説明: フリー・テキスト・ストリングの構文が正しくありません。

処置: 構文仕様に従ってフリー・テキスト・ストリングを書き換えてください。

CTE0168 **セクション・ステートメントの左括弧がありません。**

説明: セクション・ステートメントの構文が正しくありません。

処置: 構文仕様に従ってセクション・ステートメントを書き換えてください。

CTE0169 **セクション・ステートメントのコンマまたは右括弧がありません。**

説明: セクション・ステートメントの構文が正しくありません。

処置: 構文仕様に従ってセクション・ステートメントを書き換えてください。

CTE0170 **右二重引用符がありません。**

説明: 指定した検索項目の構文が正しくありません。

処置: 構文仕様に従って検索項目を書き換えてください。

CTE0171 **セクション名の左二重引用符がありません。**

説明: セクション・ステートメントの構文が正しくありません。

処置: 構文仕様に従ってセクション・ステートメントを書き換えてください。

CTE0172 **セクション名の右二重引用符がありません。**

説明: セクション・ステートメントの構文が正しくありません。

処置: 構文仕様に従ってセクション・ステートメントを書き換えてください。

CTE0173 **エスケープ文節には 1 つのエスケープ文字を定義する必要があります。**

説明: エスケープ文節で複数の文字を定義することはできません。

処置: エスケープ文節から余分な文字を除去してください。

CTE0174 **空白文字はエスケープ文字として許可されません。**

説明: エスケープ文節に空白文字を指定することはできません。

処置: エスケープ文節を、有効な文字を指定した文節に変更してください。

CTE0175 **エスケープ文節が定義されていますが、検索句にマスク文字がありません。**

説明: エスケープ文節が、マスク文字を使用せずに指定されました。

処置: エスケープ文節を除去してください。

CTE0176 **句内のエスケープ文字の後続文字が、同じ文字でないか、マスク文字ではありません。**

説明: エスケープ文字の後の文字は、マスク文字またはエスケープ文字自身でなければなりません。

処置: 検索ストリングを変更してエスケープ文字を正しく使用してください。

CTE0177 **数値 "%1" が無効です。**

説明: 検索引き数に指定された数値が無効です。

処置: 文書の有効範囲を確認してください。検索引き数の値を書き換えてください。

CTE0178 **ファジー句のマスク文字の前にはエスケープ文字が必要です。**

説明: マスキングをファジー検索と併用することはできません。

処置: 検索ストリングをエスケープ文字で更新してください。

CTE0179 **シソーラス名 "%1" が許可される長さの "%2" を超えています。**

説明: 60 バイトより長い主キーは、サポートされていません。

処置: 表のレイアウトを変更してから、再度、索引を作成してください。

CTE0180 **シソーラス "%1" が見つかりません。**

説明: 指定したシソーラスが見つかりません。

処置: シソーラス・ファイルがシソーラス・ディレクトリに配置されているか、完全に修飾されているかチェックしてください。

CTE0181 ライブラリー "%1" をロードできません。

説明: ライブラリーが見つかりません。

処置: ライブラリーがライブラリー・パスに配置され、使用可能であることがチェックしてください。DB2 を開始して停止し、現行設定値が使用されていることを確認してください。

CTE0182 関数 "%1" をライブラリー "%2" からロードできません。

説明: ライブラリー・エントリー・ポイントがロードできません。

処置: アクセスされたライブラリーが無効のようです。ライブラリーが一度だけ指定されているかチェックしてください。

CTE0183 共有システム・リソースの使用でエラーが発生しました。

説明: 共有メモリーまたはセマフォのような共有システム・リソースへの要求に答えることができません。

処置: 現行システムの状況および構成をチェックしてください。UNIX 上で `ipcs` コマンドを使用してリソースを確認してください。DB2 および DB2 Net Search Extender などのすべてのアプリケーションを停止してください。リソースがまだリストにある場合は、`ipcrm` を使用して、それらのリソースをクリーンアップしてください。

CTE0184 `db2text start` コマンドが発行されていません。

説明: ロッキングおよび更新サービスを必要とするコマンドが呼び出されました。

処置: `db2text start` を使用して、更新サービスおよびロッキング・サービスを開始してください。

CTE0185 更新およびロッキング・サービスはすでにアクティブになっています。

説明: `db2text start` が発行されましたが、更新サービスおよびロッキング・サービスはすでに実行中です。

処置: これ以上の処置はありません。

CTE0186 更新およびロッキング・サービスでエラーが発生しました。詳細については `db2diag.log` を確認してください。

説明: 更新およびロッキング・サービス・エラーが発生しました。

処置: 詳細に関して `db2diag.log` をチェックするか、共有リソースをクリーンアップしてください。CTE0183 も参照してください。

CTE0187 更新およびロッキング・サービスがまだアクティブになっています。FORCE オプションを使用してサービスを停止させてください。

説明: `db2text stop` コマンドは、ロッキング・サービスを停止しておらず、引き続き処理が実行中です。

処置: `db2text control` を使用してどの処理が実行中かチェックし、それらの処理が終了するのを待ってください。処理を停止する必要がある場合は、FORCE オプションを使用します。

CTE0188 更新およびロッキング・サービスの使用で一時的な問題が発生しました。やり直してください。

説明: `db2text stop` コマンドが、ロッキング・サービスを停止していません。プログラムが引き続き実行中であるか、矛盾する状態が検出されました。

処置: `db2text control` を使用してどの処理が実行中かチェックし、それらの処理が終了するのを待ってください。処理を停止するためには、FORCE オプションを使用してください。

CTE0189 実行可能プログラム "%1" が見つかりません。

説明: プログラム・ファイルを検出、またはアクセスできません。

処置: プログラム・ファイルが DB2 サーバーの `bin` または `adm` ディレクトリーに配置されているかどうかチェックしてください。ファイルが検出できない場合は、システムが壊れています。

CTE0190 実行可能プログラム "%1" を開始できません。

説明: プログラムを開始できません。

処置: プログラムが DB2 サーバーの `bin` または `adm` ディレクトリーに配置されており、適切なライブラリーがインストールされているかチェックしてください。詳

細に関しては、サーバー上でプログラムを手動で呼び出してください。

CTE0191 索引のドロップ操作が完了していません。
詳細に関しては **db2diag.log** をチェックしてください。

説明: 索引のドロップ操作が完了していません。
FORCE オプションが原因と思われます。

処置: FORCE オプションを使用して、エラーに無関係にすべてをドロップします。ペンディング・ファイルの索引ディレクトリーをチェックして、それらを手動で除去してください。

CTE0192 索引の更新操作でエラーが発生しました。
詳しくは、イベント表 "%1"."%2" と **db2diag.log** を確認してください。

説明: 索引更新処理中に文書エラーがイベント表に書き込まれました。

処置: 文書エラーについての詳細は、イベント表をチェックしてください。問題を修正した後、イベント・ログをクリーンアップしてください。

CTE0194 列 "%2" のタイプ "%1" はサポートされていません。

説明: サポートされた列のリストにない列が使用されました。

処置: 索引作成をチェックして、キーおよび索引付けの有効な列リストを確認してください。コマンドを適切に変更して再試行してください。

CTE0195 "%1" は絶対パスではありません。

説明: サーバー上の絶対パスが必要です。

処置: パスを確認してコマンドに絶対パスを書き込んでください。

CTE0198 対応するテキスト索引がありません。

説明: 列にテキスト索引がありません。

処置: テキスト索引がまだ存在しているかどうかを確認してください。

CTE0199 表 "%2" の列 "%1" に対応するテキスト索引がありません。

説明: テキスト索引のない列を検索しようとしてしました。

処置: 検索する列をチェックするか、列にテキスト索引を作成してください。

CTE0200 最低 1 つのコマンド・オプションを指定する必要があります。

説明: ALTER INDEX コマンドで、更新オプションおよび保管オプションのような索引の特性を変更できません。変更する特性が 1 つも指定されませんでした。

処置: 少なくとも 1 つのコマンド・オプションを指定します。すべての可能なオプションについては、コマンド構文を参照してください。

CTE0201 同じ列の、既存のテキスト索引と矛盾しません。

説明: 同じ列に定義されたテキスト索引が、この索引作成コマンドとは異なるパラメーターによって作成されています。

処置: 索引作成コマンドのパラメーター値を訂正してください。以下のパラメーターが、既存の索引と作成する索引で同じ値をもつようにしてください: ccsid、言語、フォーマット、文書モデル、索引構成、列関数、および属性。

CTE0202 オブジェクト "%1"."%2" は、キー列が指定されている場合、ビューである必要があります。

説明: 指定したオブジェクトはビューではありません。KEY COLUMNS FOR INDEX ON VIEW 文節を使用できるのは、ビューの列に索引を作成する場合だけです。

処置: KEY COLUMNS FOR INDEX ON VIEW(SQL-columnname-list) 文節を除去してください。

CTE0203 テキスト索引 "%1"."%2" が **CACHE TABLE** オプションで作成されていません。このオプションはコマンドの実行に必要です。

説明: 指定した索引が CACHE TABLE オプションを指定して作成されていない場合は、このコマンドを実行できません。

処置: CACHE TABLE オプションを指定して索引を作成してください。コマンド構文については、資料を参照してください。

CTE0204 属性名がありません。属性表現に "**AS <属性名>**" を追加してください。

説明: 属性表現で列表現を使用するときは必ず、属性名を指定する必要があります。例: (C1+C2 AS myname)。

エラー・メッセージ

処置: 属性表現に "AS <属性名>" を追加してください。

CTE0205 CACHE TABLE 式が無効です。

説明: キャッシュ表式の列リストが無効です。

処置: 索引作成コマンドのキャッシュ表列リストを訂正してください。指定した表に列が存在することを確認してください。列に関数が適用されている場合は、関数が正しく使用されていることを確認してください。

CTE0206 ATTRIBUTE 式が無効です。

説明: 属性式の列リストが無効です。

処置: 索引作成コマンドの属性列リストを訂正してください。指定した表に列が存在することを確認してください。列に関数が適用されている場合は、関数が正しく使用されていることを確認してください。

CTE0207 KEY COLUMNS FOR INDEX ON VIEW がビュー "%1"."%2" の索引に指定されていません。

説明: ビューに関する索引を作成した場合は、KEY COLUMNS FOR INDEX ON VIEW(SQL-columnname-list) 文節を指定する必要があります。列名のリストは、ビュー内の行を「一意に」識別する列を指定します。

処置: 索引作成コマンドに KEY COLUMNS FOR INDEX ON VIEW(SQL-columnname-list) 文節を組み込んでください。

CTE0208 INITIAL SEARCH RESULT ORDER 列が無効です。

説明: INITIAL SEARCH RESULT ORDER(SQL-order-by list) 式の列リストが無効です。

処置: 索引作成コマンドで列リストの ORDER BY を訂正してください。構文が正しく、指定した表に列が存在するかどうかをチェックしてください。列に関数が適用されている場合は、関数が正しく使用されていることを確認してください。

CTE0209 属性列 "%2" のタイプ "%1" はサポートされていません。タイプ DOUBLE が必要です。

説明: 属性列の場合、DOUBLE のデータ・タイプのみがサポートされています。

処置: 索引を作成するテキスト列を持つ表の属性列のタイプが DOUBLE になるようにしてください。属性列の式でキャスト演算子を使用してもいい場合があります。

DOUBLE にキャストできるデータ・タイプについては、SQL リファレンスを参照してください。

CTE0210 索引構成パラメーター "%2" の値 "%1" は無効です。有効な値は "%3" です。

説明: 構成パラメーターに指定した値が誤っています。パラメーターの有効値については、コマンド構文を参照してください。

処置: 索引作成コマンドの索引構成パラメーター値を訂正してください。

CTE0211 "%1" は無効な索引構成パラメーターです。

説明: 索引構成オプションが不明です。

処置: 索引作成コマンドの構文を確認してください。有効な索引構成オプションは、TreatNumbersAsWords および IndexStopWords です。これらのオプションは次のようにコンマで区切る必要があります。index configuration(treatnumberaswords 1, indexstopwords 1)

CTE0212 内部索引構成ファイル "%1" を保管できませんでした。

説明: 索引の内部構成ファイルを保管できませんでした。

処置: インスタンス所有者が、ファイルを保管すべきディレクトリーの書き込み許可をもっていることを確認してください。同一の名前をもったファイルがすでに存在している場合は、インスタンス所有者がそのファイルに書き込めることを確認してください。

CTE0213 内部索引構成ファイルのテンプレート "%1" をロードできませんでした。

説明: 内部索引構成ファイルのテンプレートを読み取ることができませんでした。

処置: ファイルが正しいロケーションにあり、読み取り可能であることを確認してください。

CTE0214 索引構成ファイルに新規項目 "[%1],%2=%3" を設定しようとして内部エラーが発生しました。

説明: 索引の内部構成ファイルを作成した際の内部エラー。

処置: ファイルが存在する場合は、インスタンス所有者がファイルの読み取りおよび書き込みを行えることをチェックしてください。ファイルが置かれている装置に十分なスペースがあることをチェックしてください。

CTE0215 別名 "%1"."%2" の索引作成はサポートされていません。代わりに基本表 "%3"."%4" を使用してください。

説明: 別名に索引を作成することはできません。

処置: 基本表を指定して索引作成コマンドを入力してください。

CTE0217 スケジュール・サービスはすでにアクティブです。

説明: サービスはすでにアクティブになっており、開始する必要はありません。

処置: アクションは不要です。

CTE0218 関数 "%1" がエラー・コード "%2" で失敗しました。

説明: Windows 関数は示されたエラー・コードで失敗し、処理を続けることはできません。

処置: ここに示された Windows システム・エラー・コードを使用して、詳細なエラー情報を入手してください。

CTE0219 サービス "%1" をオープンできませんでした。エラー・コードは "%2" です。

説明: 指定したサービスが、Windows システムにありません。

処置: 指定したサービスが Windows システムにインストールされているかを確認してください。ここに示された Windows システム・エラー・コードを使用して、詳細なエラー情報を入手してください。

CTE0220 DB2 インスタンス・プロファイルのパスが見つかりませんでした。

説明: DB2 インスタンス・プロファイル・パスを取得する内部 DB2 関数が失敗しました。

処置: インスタンス・プロファイル・パス情報を指定せずに DB2 インスタンスを作成し、コマンドを再試行してください。

CTE0221 UpdateFrequency "%1" は誤って指定されています。

説明: 更新頻度ステートメントの構文が正しくありません。

処置: 構文規則に従って更新頻度ステートメントを訂正してください。

CTE0222 スケジュール・サービス入力ファイル "%1" が壊れています。

説明: 索引更新情報を含むスケジューラー・ファイルが壊れています。

処置: システム・エディターを使用して問題を訂正してください。切り捨てられた項目があるか、終了改行文字が削除されている可能性があります。訂正してもファイルの内容をリストアできない場合は、以下のようにしてください。

- コマンド db2text stop を呼び出してスケジューラーを停止する。
- スケジューラー・サービス・ファイルを削除する。
- コマンド db2text start を呼び出してスケジューラーを開始する。
- コマンド db2text alter index ... を使用して該当するすべての索引の更新頻度項目を再作成する。

CTE0223 ファイル "%1" をクローズできませんでした。

説明: 指定したファイルがクローズできません。

処置: ファイルが正しく指定されていることを確認してください。

CTE0224 ファイル "%1" を "%2" にコピーできませんでした。

説明: 最初のファイルを 2 番目のファイルにコピーできません。

処置: ファイルが正しく指定されていることを確認してください。2 番目のファイルが存在しており、読み取り専用であることを確認してください。システムに十分なフリー・スペースがあることも確認してください。

CTE0225 ファイル "%1" を除去できませんでした。

説明: 指定したファイルがシステムから除去できません。

処置: ファイルが正しく指定されていること、およびファイルのアクセス権限を確認してください。

CTE0225 ファイル "%1" を除去できませんでした。

説明: 指定したファイルがシステムから除去できません。

処置: ファイルが正しく指定されていること、およびファイルのアクセス権限を確認してください。

CTE0227 ファイル "%1" への書き込み操作が失敗しました。

説明: 指定したファイルが書き込み可能ではありません。

処置: ファイルが正しく指定されていること、およびファイルのアクセス権限を確認してください。システムに十分なフリー・スペースがあることも確認してください。

CTE0228 ユーザーにオペレーティング・システム・レベルでの十分なアクセス権限がありません。

説明: そのコマンドには、オペレーティング・システム・レベルでの管理者権限が必要です。

処置: オペレーティング・システムの管理者権限があることを確認してください。管理者グループのメンバーであるかどうかを確認してください。

CTE0231 "%1" は、"%2"."%3" の表スペースと同じノード・グループ ("%4") に定義されていません。

説明: 索引が作成されるテキスト列を含む表と全く同じように、管理表の表スペースを異なるノードに分散する必要があります。これが必ず行えるように指定した表スペースが同じノード・グループで定義されているかどうかチェックされます。

処置: 索引が作成されるテキスト列を含む表と同じノード・グループで定義される表スペースを指定してください。

CTE0232 指定またはデフォルトの表スペース "%1" は単一ノードではありません。これは、ビューの索引か、**CACHE TABLE** オプションが指定された場合に必要です。

説明: 単一ノード上の表の場合に限り、ビューの索引または **CACHE TABLE** オプションを使用可能にすることがサポートされます。

処置: デフォルトの表スペースがこのエラーの原因になる場合は、単一ノードの表スペースに表を置いてください。あるいは、別の複数ノードの表スペースを指定した場合は、単一ノードの別の表スペースを指定してください。

CTE0233 実行中の管理コマンドに競合があります。このコマンドを後からやり直してください。

説明: 別の管理コマンドがまだ実行中であるか、コマンド・ロックを解放せずに異常終了しています。

処置: **CONTROL LIST** をチェックして、どのロックがアクティブになったままであるかを確認してください。アクティブなロックはあるが、コマンドが実行されていない場合は、**CONTROL CLEAR** コマンドを使用してロックを手動でクリアしてください。ロックを保持する管理コマンドを実行しているユーザーが他にいる可能性があるので注意してください。

CTE0234 テキスト索引に関して実行中の、競合する管理コマンドがあります。このコマンドを後でやり直すか、または **DISABLE DATABASE** コマンドの **FORCE** オプションを指定してください。

説明: 別の管理コマンドがまだ実行中であるか、コマンド・ロックを解放せずに異常終了しています。

処置: **CONTROL LIST** をチェックして、どのロックがアクティブになったままであるかを確認してください。アクティブなロックはあるが、コマンドが実行されていない場合は、**CONTROL CLEAR** コマンドを使用してロックを手動でクリアしてください。ロックを保持する管理コマンドを実行しているユーザーが他にいる可能性があるので注意してください。 **DISABLE DATABASE** コマンドの場合、そのデータベース上のその他のすべてのコマンドを停止する **FORCE** オプションを指定できません。

CTE0235 **DB2 Net Search Extender** の有効なライセンスがありませんでした。

説明: **DB2 Net Search Extender** の有効なライセンスがありませんでした。

処置: ライセンスが **db2lic** を使用して正しくインストールされているかチェックしてください。製品のインストール後に既存インスタンスが更新されていることを確認します。

CTE0236 **MPP** インスタンスでは、**Node0** のみがサポートされています。

説明: 索引が作成されるテキスト列をもつ表が **Node0** 上にある場合、**MPP** インスタンスに関してのみテキスト索引を作成できます。

処置: 表が定義されている表スペースのノード・グループを確認してください。

CTE0237 内部エラー: ログ表 "%1"."%2" には無効な操作 "%3" が含まれています。

説明: ログ表は、索引が作成されるテキスト列を含む表で実行される操作を追跡します。この表に DB2 Net Search Extender が作成しなかった項目が含まれており、この表が壊れている可能性があります。

処置: ログ表をチェックし、壊れた項目を削除してください。

CTE0238 内部エラー: 表 "%1"."%2" の列 "%3" に、誤った構文式が含まれています。

説明: 指定したテキスト列の式リストにエラーがあります。

処置: 区切り文字の Begin と End のペアをチェックしてください。

CTE0239 内部エラー: 索引プロパティ "%1" の全長が最大長の "%2" を超過していません。

説明: 索引プロパティの最大サイズ (1016 バイト) を超えています。プロパティには、インスタンス、索引、作業ディレクトリ、およびその他の情報が含まれています。

処置: これらのパス名が長過ぎないことを確認してください。

CTE0240 内部エラー: 環境変数 "%1" の設定に失敗しました。

説明: 指定した環境変数の設定が失敗しました。環境設定に関する問題の可能性があります。

処置: ご使用の OS のガイドラインを調べてください。

CTE0241 内部エラー: データ・リンク UDF "%1"."%2" がタイプ "%3" を戻しました。予想タイプは "%4" でした。

説明: データ・リンク参照の内容の取り出しに使用されるデータ・リンク UDF が、予期しないデータ・タイプで定義されています。

処置: SYSCAT.FUNCTIONS からデータ・リンク UDF の定義を取り出し、エラーを IBM 技術員にお知らせください。

CTE0242 パラメーター "%2" の値 "%1" が無効です。

説明: 検索ストアード・プロシージャまたは表値関数 DB2EXT.TEXTSEARCH が、無効なパラメーターを指定して呼び出されました。

処置: 検索ストアード・プロシージャまたは表値関数のパラメーター値を訂正してください。有効なパラメーターについては、資料を参照してください。

CTE0243 テキスト索引 "%1"."%2" のキャッシュがアクティブになっていません。

説明: Net Search Extender の操作では、キャッシュがアクティブになっている必要があります。キャッシュは現在、アクティブではありません。考えられる理由として、以下のものがあります。

- 最後の DB2TEXT START コマンド以後、キャッシュがアクティブにされていない。
- DB2TEXT DEACTIVATE CACHE コマンドを使用してキャッシュが明示的に非アクティブにされた。

処置: 索引に対して DB2TEXT ACTIVATE CACHE コマンドを実行し、Net Search Extender 操作を再実行してください。

CTE0244 内部エラー: "%1" の呼び出しで rc="%2"、SQLCODE="%3" が戻りました。

説明: 内部関数の呼び出しの際に、内部処理エラーが発生しました。

処置: エラーが継続する場合は、トレースを開始して db2diag.log を確認してください。エラーを報告してください。

CTE0245 要求されたキャッシュ・サイズは使用可能なキャッシュ・サイズを超過しています。最大キャッシュ・サイズを値 "%1" より大きくするか、pctfree 値を減らしてください。

説明: すべてのデータをロードするのに必要なキャッシュ・サイズが、索引の MAXIMUM CACHE SIZE 値を超えています。キャッシュをアクティブ化 (DB2TEXT ACTIVATE コマンド) する際に、あるいはキャッシュがアクティブ化されているときの索引更新操作によって、このエラーが検出されることがあります。

処置: DB2EXT ACTIVATE コマンドでエラーが報告される場合は、DB2EXT.MAXIMUM_CACHE_SIZE 関数を使用して最大キャッシュ・サイズを再計算し、索引の MAXIMUM CACHE SIZE 設定を変更してください。最

エラー・メッセージ

最終的に PCTFREE 値を小さくします。インクリメンタル更新の際に文書の最大数を超える場合は、コマンド db2 deactivate cache および db2text activate cache recreate を使用してキャッシュを再作成してください。

CTE0246 ファイル "%1" が空です。

説明: コマンドで指定した文書モデル・ファイルが空のため、DB2TEXT CREATE INDEX コマンドが失敗しました。

処置: コマンドで有効な文書モデルを指定してください。

CTE0247 DB2 Net Search Extender のストアード・プロシージャを作成できませんでした。

説明: DB2TEXT ENABLE DATABASE コマンドで内部ストアード・プロシージャ DB2EXT.CTESRVSP を作成するのに失敗しました。

処置: 詳細については、CREATE PROCEDURE ステートメントに関連した追加の DB2 エラー・メッセージで確認してください。同じ名前の既存ストアード・プロシージャを除去してもエラーが訂正できない場合は、トレースを開始し、エラーを報告してください。

CTE0248 生成された検索ストリングが長すぎます。検索照会を簡単なもの書き換えてください。

説明: Net Search Extender 照会が長すぎるか複雑過ぎて、基本検索エンジンが処理できません。複雑さは、シソーラスの拡張、FUZZY FORM OF 式、およびマスク文字に影響されます。

処置: 照会をより単純なものにするか、照会の長さを短くしてください。

CTE0249 実行可能プログラム "%1" は異常終了しました。

説明: Net Search Extender コマンドの実行中に、実行可能プログラム "%1" が呼び出されましたが、異常終了しました。

処置: 実行可能プログラムが、ユーザー対話 (たとえば、シグナル) によって明示的に終了されなかったか確認してください。されていない場合、トレースを開始し、コマンドを再実行して、エラーを報告してください。

CTE0250 列タイプ・トランスフォーメーション機能 "%2"."%3" のリターン・タイプ "%1" はサポートされていません。

説明: DB2TEXT CREATE INDEX コマンドで、サポートされていないデータ・タイプを戻す列タイプ変換が指定されました。サポートされているデータ・タイプは次のとおりです。CHARACTER、VARCHAR、LONG VARCHAR、CLOB、GRAPHIC、VARGRAPHIC、LONG VARGRAPHIC、DBCLOB、BLOB、DATALINK。

処置: 別の列タイプ変換関数を選択してください。

CTE0251 内部エラー: 列タイプ "%1" はサポートされていません。

説明: サポートされているタイプのリストにない列タイプが使用されています。

処置: 索引作成をチェックして、キーおよび索引付けの有効な列リストを確認してください。コマンドを適切に変更して再試行してください。エラーが継続する場合は、トレースを開始して、db2diag.log もチェックしてください。エラーを IBM 技術員にお知らせください。

CTE0252 パラメーター "%1" がありません。

説明: 内部エラー - Net Search Extender コマンドを実行する際に管理実行可能プログラムが呼び出されましたが、パラメーター "%1" が指定されていません。

処置: Net Search Extender パラメーター・コマンドを変更して、問題を回避してください。エラーが継続する場合は、トレース機能をオンにして、エラーを IBM 技術員にお知らせください。

CTE0253 ログ・ビューにリストされた文書が見つかりません。

説明: ログ・ビューにリストされているテキスト文書の内容が変更されており、アクセスできませんでした。

処置: 文書が存在すること、およびテキスト文書の読み取り/アクセス許可が索引に組み込まれていることを確認してください。

CTE0254 索引 "%1" のキャッシュは、既にアクティブになっています。

説明: ACTIVATE CACHE コマンドを使用して、索引がすでにアクティブ化されています。

処置: 指定した索引名および使用しているデータベースを確認してください。

CTE0255 キャッシュ結果列式の列名がありません。式に "AS <cache column name>" を追加してください。

説明: キャッシュ結果列式に名前を付ける必要があります。例: 'C1+C2 AS myresult'。

処置: 式に "AS <cache column name>" を追加してください。

CTE0256 索引付け用のデータの選択に必要な照会が失敗しました。属性、キャッシュ表または検索結果順序式の複雑さを削減してください。

説明: データベースから索引を作成するデータを選択するために、Net Search Extender はコマンドの式から照会を作成します。複雑過ぎて照会が失敗しました。

処置: 属性、キャッシュ表、または初期検索結果順序の式をより単純なものにしてください。

CTE0257 共有メモリーの作成エラーです。

説明: 以前のエラーまたはアクセス権の問題によって、共有メモリー・リソースが作成できませんでした。

処置: 詳細に関しては、db2diag.log をチェックするか、共有リソースをクリーンアップしてください。エラー CTE0183 も参照してください。

CTE0258 共有メモリーのバージョン・エラーです。

説明: 壊れているかバージョンに矛盾があるため、共有メモリー・リソースにアクセスできませんでした。

処置: 詳細については db2diag.log をチェックしてください。データベースを使用不可にしてから、再び使用可能にし、もう一度試行してください。

CTE0259 グローバル共有メモリーに項目を挿入できません。項目がすでに存在しています。

説明: 以前のエラーのため、グローバル共有メモリーに挿入する項目がすでに存在しています。

処置: 詳細については db2diag.log をチェックしてください。コマンド db2text stop および db2text start を使用して、更新サービスおよびロッキング・サービスを再始動してください。

CTE0260 グローバル共有メモリー内の項目にアクセスできません。項目が見つかりません。

説明: 以前のエラーのため、グローバル共有メモリーから除去する項目が存在しません。

処置: 詳細については db2diag.log をチェックしてください。コマンド db2text stop および db2text start を使用して、更新サービスおよびロッキング・サービスを再始動してください。

CTE0261 このインスタンスに、テキスト索引に対してアクティブになっているキャッシュが、少なくとも 1 つあります。DEACTIVATE CACHE コマンドを使用して、アクティブ化された索引のキャッシュを非アクティブ化するか、FORCE オプションを使用して停止してください。

説明: ACTIVATE CACHE コマンドを使用してアクティブ化されているすべてのテキスト索引に対して、DEACTIVATE CACHE コマンドを実行する場合のみ、db2text stop コマンドを使用できます。

処置: DEACTIVATE CACHE コマンドを使用してアクティブ化されている索引のキャッシュを非アクティブ化するか、FORCE オプションを使用して停止してください。

CTE0262 パラメーター "%1" の値が長すぎます。

説明: 値が許容最大サイズを超えています。

処置: 最大サイズをチェックしてください。

CTE0263 テキスト索引 "%1"."%2" は RECREATE INDEX ON UPDATE オプションで作成されました。このコンテキストでは、UPDATE MINIMUM または COMMITCOUNT FOR UPDATE を指定することはできません。

説明: 索引を増大して更新する場合のみ、UPDATE MINIMUM および COMMITCOUNT FOR UPDATE が有効になります。

処置: 更新が行われるたびに索引を再作成したい場合は、UPDATE MINIMUM および COMMITCOUNT FOR UPDATE の設定を除去してください。UPDATE MINIMUM および COMMITCOUNT FOR UPDATE を使用したい場合は、RECREATE INDEX ON UPDATE を指定しないでください。

CTE0264 索引のアクティブ化操作でエラーが発生しました。詳しくは、イベント・ビュー "%1"."%2" と db2diag.log を確認してください。

説明: 索引のアクティブ化処理中に、イベント表および db2diag.log ファイルにエラーが書き込まれました。

エラー・メッセージ

処置: 文書エラーについての詳細は、イベント表をチェックしてください。問題を修正した後、イベント・ログをクリーンアップしてください。

CTE0265 ユーザー表の表スペースまたは管理表スペース ("**%1**") が、ノード **0** 以外にも定義されています。

説明: MPP インスタンスに関してテキスト索引を作成する場合は、ユーザー表の表スペースは Node0 以外にあってはなりません。

処置: 表スペースが Node0 に存在する表を使用してください。

CTE0266 ValueFrom "**%1**" は ValueTo "**%2**" よりも小さくなければなりません。

説明: 属性検索で指定した値が無効です。検索構文が 'BETWEEN ValueFrom AND ValueTo' である場合は、下方境界 (ValueFrom) が上方境界 (ValueTo) より小さくなければなりません。

処置: 'BETWEEN ValueFrom AND ValueTo' 文節の境界を変更してください。

CTE0267 データベース "**%1**" 内の Net Search Extender データベース・オブジェクトは、矛盾する状態にあります。

説明: 少なくとも 1 つの DB2 Net Search Extender オブジェクトが欠落しているか、または壊れています。

DB2 Net Search Extender 製品の新規バージョンのインストール後にデータベースが移行されていない、またはデータベース・ユーザーが Net Search Extender の内部オブジェクト (複数の場合あり) を変更またはドロップしました。この場合、すべてのテキスト索引は失われるので、データベースはテキストでの使用を不可にする必要があります。

処置: 現行バージョンへのデータベースの移行については、DB2 Net Search Extender 文書にある移行の説明に従ってください。または、FORCE オプションを使用して DB2TEXT DISABLE DATABASE コマンドを発行してください。その後、DB2TEXT ENABLE DATABASE コマンドを使用して、データベースをテキストのために使用可能にすることができます。

CTE0270 ログ表 "**%1**."**%2**" は、インクリメンタル更新後に変更できませんでした。項目は次の UPDATE で処理されます。

説明: ある特定の状態では、トランザクション内での変更が、更新の開始時点では非コミットされているのに、索引更新の実行中にコミットされることがあります。

これは、潜在的に矛盾につながる可能性があります。

このような矛盾する状態を避けるために、しきい値のタイム・スタンプより前の変更レコードは、部分的に処理されていても、ログ表から削除されません。次のインクリメンタル更新で、変更は索引に再度適用されます。

処置: 次の索引更新で、変更は索引に再度適用されません。削除操作の場合は、これによって以下のエラーが発生する場合があります。 CTE0101:

ItlEnReasonCode_Docmap_docid_not_found.

文書はすでに削除されているので、このエラーは無視してかまいません。 CTE0270 エラーが頻繁に発生する場合は、索引をドロップし、インクリメンタル索引更新用のタイム・スタンプしきい値を変更して、索引を再作成することを検討してください。例: db2text "CREATE INDEX ... INDEX CONFIGURATION(UPDATEDELAY 30)"

これが意味するのは、インクリメンタル更新の実行中は、30 秒を超過している古いレコードだけを変更し、並行して実行される 30 秒未満の変更トランザクションを妨害しないようにするということです。

CTE0273 索引 "**%1**","**%2**" のキャッシュはすでにアクティブ化されています。

説明: ACTIVATE CACHE コマンドを使用して、索引がすでにアクティブ化されています。

処置: 指定した索引名および使用しているデータベースを確認してください。

CTE0274 この接続のターゲット・データベース・システム "**%1**" はサポートされていません。

説明: DB2 Net Search Extender がサポートしていないデータベース・システムへの接続で、DB2TEXT コマンドを実行しようとした。

CTE0275 サーバー "**%1**" のタイプおよびバージョン情報が見つかりませんでした。

説明: サーバーのタイプおよびバージョン情報が、DB2 カタログ・ビュー 'SERVERS' で見つかりませんでした。

処置: DB2 のフェデレーテッド環境が正しくセットアップされていることを確認してください。

CTE0276 データ・リンク UDF が見つかりません
でした。

説明: データ・リンク UDF がデータベースにユーザー定義関数として登録されていません。

処置: ディレクトリー db2ext/ddl でコマンド 'db2 -tvf ctedlcud.ddl' を実行してください。

CTE0277 キャッシュ・メモリー・セグメントがア
ツチできませんでした。

説明: システムが、ラージ・キャッシュ・セグメントのロードに十分なメモリーを割り振れません。またはキャッシュ・セグメントが事前に削除されているためにオープンできません。

処置: システム設定を確認して、ページング・スペースの量を増やし、メモリーを解放してください。キャッシュ・サイズが大きい場合は、システムの準備が必要な場合があります。DB2 Net Search Extender 文書を参照してください。キャッシュを再作成するには、DEACTIVATE コマンドおよび ACTIVATE [RECREATE] コマンドを使用してください。問題が解決しない場合は、db2diag.log の追加情報を確認してください。

CTE0278 AIX の 32 ビット・システムでは、
MAXDATA 設定を変更してから、ラ
ージ・キャッシュをアクティブ化してくだ
さい。

説明: AIX 32 ビット・システム上でストアード・プロシージャの検索を使用する場合は、db2fmp 実行可能ファイルの MAXDATA 設定を変更する必要がある場合があります。

処置: MAXDATA 設定の変更についての詳細は、DB2 Net Search Extender 文書を参照してください。

CTE0279 キャッシュ・データのサイズが、システム
しきい値に達しました。

説明: PCTFREE 値を減らすことにより、キャッシュのアクティブ化中のデータの最大サイズを増やすことができます。これにより、システムは、キャッシュ内のフリー・スペースの予約を減らすことができます。

処置: より低い PCTFREE 値を使用するか、または、キャッシュされるテキスト・データの量を削減してください。キャッシュを再作成するには、DEACTIVATE コマンドおよび ACTIVATE [RECREATE] コマンドを使用してください。

CTE0280 永続キャッシュ・ファイルを書き込むため
のディスク・スペースが不足しています。

説明: システムは、キャッシュ・ディレクトリーに、永続キャッシュに十分な大きさのファイルを書き込めません。

処置: ALTER INDEX コマンドを使用して、永続キャッシュのディレクトリーを空のファイル・システムに変更してください。あるいは、PCTFREE 値または MAXIMUM CACHE SIZE 値を減らすか、一時キャッシュを使用して、キャッシュ・サイズを削減してください。

CTE0281 永続キャッシュ・ファイル "%1" の削除
に失敗しました。

説明: 当該ファイルが存在しないか、またはアクセスできません。

処置: このファイルがまだ存在するかどうかを確認して、手動で削除してください。

CTE0282 キャッシュ内の文書の数が、システムしき
い値に達しました。

説明: PCTFREE 値を減らすことにより、キャッシュのアクティブ化中にキャッシュされる文書項目の最大数を増やすことができます。これにより、システムは、キャッシュ内のフリー・スペースの予約を減らすことができます。

処置: より低い PCTFREE 値を使用するか、または、キャッシュ内の文書項目の量を削減してください。キャッシュを再作成するには、DEACTIVATE コマンドおよび ACTIVATE [RECREATE] コマンドを使用してください。

CTE0283 キャッシュ・メモリー・セグメントを作成
できませんでした。

説明: システムは、ラージ・キャッシュ・セグメントをメモリー内にロードするのに十分なメモリーを割り振れません。PCTFREE 値を減らすことにより、キャッシュ・セグメントのサイズをさらに小さくすることができます。

処置: システム設定を確認して、ページング・スペースの量を増やし、メモリーを解放してください。より低い PCTREE 値を使用することにより、キャッシュ・サイズを減らすこともできます。キャッシュ・サイズが大きい場合は、システムの準備が必要な場合があります。DB2 Net Search Extender 文書を参照してください。キャッシュを再作成するには、DEACTIVATE コマンドおよび ACTIVATE [RECREATE] コマンドを使用してくだ

エラー・メッセージ

さい。問題が解決しない場合は、db2diag.log の追加情報を確認してください。

CTE0284 テキスト索引はノード "%1" 上にありますが、探索関数はノード "%2" 上で呼び出されました。

説明: ストアード・プロシージャの探索または表値関数 DB2EXT.TEXTSEARCH が、索引が配置されているノード上で呼び出されませんでした。この検索関数は、正しいノードへ自動的に配布されません。

処置: データベースに接続する前に、索引が配置されているノードに DB2NODE 環境変数を設定してください。

CTE0285 複数のノードに分散しているテキスト索引に対しては、検索関数は許可されません。

説明: 表値関数 DB2EXT.TEXTSEARCH は、正しいノードに自動的に配布されず、コーディネーター・ノード上で実行されるため、複数のノードに分散している索引については呼び出してはなりません。

処置: 複数ノード環境では CONTAINS、SCORE、または NUMBEROFMATCHES 関数を使用してください。

CTE0286 ソース表 "%2"."%3" およびキャプチャー変更表 "%4"."%5" について、"%1"."IBMSNAP_REGISTER" に行が見つかりませんでした。

説明: DB2TEXT CREATE INDEX コマンドで指定されたレプリケーション・キャプチャー表の特性について、IBMSNAP_REGISTER 表に有効な項目が見つかりませんでした。有効な項目は、SOURCE_VIEW_QUAL=0 を指定された、列 SOURCE_OWNER および SOURCE_NAME の索引について指定されたソース表、および列 PHYS_CHANGE_OWNER および PHYS_CHANGE_TABLE について指定されたレプリケーション・キャプチャー表を含んでいる必要があります。

考えられる原因: 指定されたソース表が、そのレプリケーション・キャプチャー表のレプリケーション・ソースとして登録されていません。

処置: そのソース表を DB2 レプリケーション用に正しく登録するか、またはそのソース表の正しいレプリケーション・キャプチャー表を指定してください。

CTE0287 ソース表 "%4"."%5" およびキャプチャー変更表 "%6"."%7" について、"%3"."IBMSNAP_REGISTER" の "%2" の値 "%1" が無効です。

説明: IBMSNAP_REGISTER 表で見つかったレプリケーションの設定は、許可されていません。考えられる原因: 1 列 CHG_UPD_TO_DEL_INS は、値 'Y' を含んでいない。2 列 CCD_CONDENSED は値 'Y' を含んでいる。

処置: ソース表を DB2 レプリケーション用に登録する際、更新操作が削除と挿入操作のペアにトランスフォームされることを確認します。さらに、コンデンス・レプリケーション・キャプチャー表が使用されないことを確認します。

CTE0288 ソース表 "%1"."%2" とキャプチャー変更表 "%3"."%4" が、別のサーバー ("%5" と "%6") 上にあります。

説明: 指定されたソース表とレプリケーション・キャプチャー表は、同じサーバー上にある必要があります。

CTE0289 ラッパー "%1" がサポートされていません。

説明: そのラッパーはサポートされていません。サポートされるラッパーのリストについては、DB2 Net Search Extender の文書を参照してください。詳細については、128 ページを参照してください。

CTE0290 別名 "%1"."%2" は、レプリケーション文節では許可されません。

説明: レプリケーション文節では、ニックネームの別名の指定は許可されていません。

処置: 別名の代わりにニックネームを指定するか、またはそのリモート表用に新しいニックネームを作成します。

CTE0360 特定のエラー・メッセージが表示される

説明: このメッセージ番号は、特定の Net Search Extender エラー用に予約済みです。

処置: 特定のエラー・メッセージへの応答。

CTE0451 指定された文書フォーマット "%1" は、強調表示 UDF ではサポートされていません。

説明: 文書フォーマット "%1" は、強調表示をサポートしていません。

処置: 強調表示 UDF がサポートしている文書フォーマットを使用してください。

CTE0452 強調表示 UDF のオプション "%1" の後に構文エラーがあります。

説明: 指定されたオプションの前後に誤った構文を指定しました。

処置: オプション %1 付近の構文を確認してください。訂正してから、再び試してください。

CTE0453 強調表示 UDF の戻りサイズが小さすぎます。

説明: 強調表示されている文書の要求された部分が、強調表示 UDF の戻りパラメーターに適合しません。

処置: ウィンドウ数、ウィンドウ・サイズ、およびまたはヒットを表示するはずのセクション数を減らしてください。これにより、ユーザーに戻される文書の部分が削減されます。

CTE0454 コード・ページ "%1" からコード・ページ UTF8 への強調表示 UDF のパラメーターの変換エラー。

説明: 指定された CCSID (これはデフォルトのデータベース CCSID の場合があります) 内の強調表示 UDF のパラメーターを UTF8 に変換できません。

処置: CCSID の正しい指定を確認してください。

CTE0455 データベースのコード・ページ "%1" は、強調表示 UDF ではサポートされていません。

説明: そのデータベースには、強調表示 UDF でサポートされていないコード・ページが含まれています。

CTE0456 強調表示 UDF は、コード・ページ UTF8 の文書のみをサポートしています。

説明: コード・ページ UTF8 の文書のみが強調表示 UDF をサポートしています。

CTE0457 パラメーター "%2" の値 "%1" は、強調表示 UDF では無効です。

説明: 強調表示されているパラメーターの値が無効です。

処置: パラメーター値をチェックして、その値がデータ範囲で許可されていることを確認します。

CTE0841 コマンド・オプション "%1" がありません。

説明: 必要なコマンド・オプションが指定されませんでした。

処置: 指定されたパラメーターを調べて、欠落しているパラメーターを追加してください。

CTE0842 コマンド・オプション "%1" の値が指定されていません。

説明: コマンド・オプションに必要な値が指定されませんでした。

処置: 指定されたパラメーターを調べて、欠落しているオプションを追加してください。

CTE0843 コマンド・オプション "%1" に数値が指定されていません。

説明: 数値ではなくストリングが指定されています。

処置: 指定されたパラメーターを調べて、ストリングを正しい数値に変更してください。

CTE0844 定義ファイル・パス "%1" が長すぎます。

説明: 指定されたパスが長すぎて、処理できませんでした。

処置: より短いパスを使用して、やり直してください。

CTE0845 定義ファイルが指定されていません。

説明: 定義ファイルを指定する必要があります。

処置: 有効な定義ファイルを追加して、呼び出しをやり直してください。

CTE0846 定義ファイル名 "%1" が長すぎます。

説明: 指定された定義ファイル名が長過ぎます。

処置: 定義ファイル名の長さを許可された長さまで削減してください。

CTE0847 定義ファイル "%1" が存在しません。

説明: 指定された定義ファイルが見つかりませんでした。

処置: 定義ファイルが正しいパスにあり、現行ユーザーがアクセス可能であることを確認してください。

CTE0849 ディクショナリー・ファイル "%1" をロックできませんでした。

説明: 処理は、ディクショナリー・ファイルをロックできませんでした。ユーザーに書き込みアクセス権限がないか、または別の処理が、書き込みのためにそのファイルをオープンしています。

処置: 実行中の処理を調べて、ディクショナリー・ファイルをロックしている処理がないことを確認し、ユーザーのアクセス権限を確認してください。

CTE0850 出力ファイル "%1" はすでに存在しません。

説明: 指定された出力ファイルを上書きできませんでした。

処置: 指定されたディレクトリーにシソーラスを作成できることを確認してください。

CTE0851 ディクショナリー・ファイル "%1" の完全性が失われました。

説明: シソーラス・ディクショナリー・ファイルが壊れています。

処置: ディレクトリーをクリーンアップし、定義ファイルをもう一度コンパイルしてください。

CTE0852 ディクショナリー・ファイル "%1" のバージョン・エラーです。

説明: ディクショナリー・ファイルが、旧バージョンのシソーラス・コンパイラーで生成されました。

処置: 現行バージョンのシソーラス・コンパイラーを使用して定義ファイルを再度コンパイルしてください。

CTE0853 既存のディクショナリー "%1" を上書きできません。

説明: 既存のディクショナリーを上書きできません。

処置: 当該ディクショナリー・ファイル、そのディレクトリー・ロケーション、およびサブディレクトリー・ロケーションに関する書き込みアクセス権限を確認してください。

CTE0855 シソーラス用語が誤って指定されています。

説明: 定義ファイルに構文エラーがあります。

処置: シソーラス定義ファイルの作成およびシソーラス・サポートについては、DB2 Net Search Extender の文書を調べてください。

CTE0856 定義ファイル "%1" が空です。

説明: 空の定義ファイルは許可されていません。

処置: シソーラス定義ファイルの作成およびシソーラス・サポートについては、DB2 Net Search Extender の文書を調べてください。

CTE0857 ファイル "%1" の行 "%2" に、ブロック開始行が見つかりません。

説明: 定義ファイルに構文エラーがあります。

処置: ブロックは、':WORDS' で始まる必要があります。シソーラスの概念については、DB2 Net Search Extender の文書を調べてください。

CTE0858 ファイル "%1" の行 "%2" に、無効なリレーションシップが指定されています。

説明: 定義ファイルに構文エラーがあります。

処置: 「associated-term-definition」を調べる必要があります。シソーラス定義ファイルの作成については、DB2 Net Search Extender の文書を調べてください。

CTE0859 ファイル "%1" の行 "%2" のリレーションシップ番号が範囲外です。

説明: ユーザー定義の関係は、すべて連想タイプに基づいています。これらは、1 から 128 の間のユニークな番号によって識別されます。

処置: リレーションシップ番号を確認してください。

CTE0861 ファイル "%1" の行 "%2" に用語が定義されていません。

説明: 必要な用語が指定されていません。

処置: シソーラス定義ファイルの作成については、DB2 Net Search Extender の文書を調べてください。

CTE0861 ファイル "%1" の行 "%2" のシソーラス用語が長すぎます。

説明: シソーラス用語の長さは、64 バイトに制限されています。

処置: シソーラス用語の長さを変更して、やり直してください。

CTE0862 ファイル "%1" の行 "%2" の強さが誤って指定されています。

説明: 定義ファイルに構文エラーがあります。

処置: シソーラス定義ファイルの作成およびシソーラ

ス・サポートについては、DB2 Net Search Extender の文書を調べてください。

CTE0863 ファイル "%1" の行 "%2" の強さが範囲外です。

説明: 強さの値は、1 から 100 までの間で指定する必要があります。

処置: 強さの値を、1 から 100 までの数値となるように変更してください。

CTE0864 内部エラー: シソーラス・コンパイラーが理由コード "%1" で失敗しました。

説明: これ以上の処理ができない内部処理エラーが発生しました。更新サービスとロッキング・サービス、および DB2 の開始および停止を試みてください。

処置: エラーが継続する場合は、トレースを開始して、db2diag.log もチェックしてください。

CTE0865 ディレクトリー "%1" を作成できませんでした。

説明: 指定されたディレクトリーを作成できませんでした。

処置: そのディレクトリーがすでに存在するかどうか、およびそのディレクトリーのアクセス権を確認してください。

CTE0866 ディレクトリー "%1" を除去できませんでした。

説明: 当該ディレクトリーを除去できませんでした。

処置: 指定されたディレクトリーの書き込み権限を持っていることを確認してください。

エラー・メッセージ

付録 G. 文書モデルの参照情報

DB2 Net Search Extender は、文書モデルの以下の参照情報を提供します。

- 文書モデル用の DTD
- ロケーター (XPath) 式のセマンティクス
- テキスト・フィールドおよび文書属性の制限
- Outside-In タグ属性値

文書モデル用の DTD

文書タイプ定義 (DTD) の形式の文書モデルの構文を正式に記述したものを次に示します。

```
<!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>
```

ロケーター (XPath) 式のセマンティクス

XML データ・モデルに従えば、XML 文書は、以下の種類のノードを含むツリーのように見えます。

- ルート・ノード
- エレメント・ノード
- テキスト・ノード
- 属性ノード
- ネーム・スペース・ノード
- 処理命令ノード
- コメント・ノード

これらのノード間のリンク、言い換えればツリー構成リレーションシップは、XML 文書の直接の包含リレーションシップを反映しています。

ルート・ノードはルートにのみ存在することができ、ツリーの他の場所には存在できません。このノードには、子として、文書エレメント、オプションのコメント、および処理命令が含まれます。

エレメント・ノードには、ルート・ノード以外のあらゆる種類のノードが含まれます。その他の種類のノードは、ツリーの端末ノードにしか、なることができません。

「子」、「属性」、および「ネーム・スペース」の、3つの種類の**包含リンク**があります。「属性」および「ネーム・スペース」の包含リンクは、それぞれ属性ノードおよびネーム・スペース・ノードにつながっている必要があります。言い換えれば、エレメント・ノードの子にアクセスするには (グラフ理論からすれば)、「属性」リンクに従ってすべての包含属性を見つけ、「ネーム・スペース」リンクに従ってすべての包含ネーム・スペース宣言を見つけ、さらに、「子」リンクに従って包含されたエレメント、テキスト・ノード、処理命令、およびコメントを見つける必要があります。

XPath 式はコンテキスト・ノードを基準にして解釈する必要があり、一連のノードを表します。Net Search Extender セレクター・パターンとして使用する場合、コンテキスト・ノードは自由です。つまり、相対パス・パターン `p` は `//p` として解釈されます。

Net Search Extender XPath セレクター・パターンを以下に示します。

- コンテキスト `N` での Pattern `'|'` LocationPathPattern は、両方ともコンテキスト `N` にある Pattern および LocationPathPattern に一致するノードの和集合を表します。
- コンテキスト `N` での `'/'` RelativePathPattern は、この RelativePathPattern がルート・コンテキストで表すすべてのものを表します。
- コンテキスト `N` での `'//'` RelativePathPattern は、ルートの子孫 (子軸上の) であるすべてのコンテキストで、この RelativePathPattern が表すと解釈されるものの和集合を表します。

- `RelativePathPattern '/' StepPattern` はコンテキスト `N` 内のノードに一致します。ただし、そのノードがその親のコンテキストでの `StepPattern` に一致し、さらに、その親ノードがコンテキスト `N` での `RelativePathPattern` に一致する場合には限られます。
- `RelativePathPattern '// StepPattern` はコンテキスト `N` 内のノードに一致します。ただし、そのノードがその親のコンテキストでの `StepPattern` に一致し、さらに、コンテキスト `N` での `RelativePathPattern` に一致する上位ノードをもつ場合に限られます。
- コンテキスト `N` での `'child'::NodeTest` (省略構文: `NodeTest`) は、`N` の子 (子軸上の) であり、`NodeTest` を満たすノードに一致します。
- コンテキスト `N` での `'attribute'::NodeTest` (省略構文: `@NodeTest`) は、`N` の属性であり、`NodeTest` を満たすノードに一致します。
- `NodeType '(' ')'` は、指定したタイプのノードの場合のみ条件を満たします。
- `'processing-instruction' '(' Literal ')'` は、リテラルを名前として持つすべての処理命令タイプのノードを満たします。
- `'*'` は、すべてのエレメント・ノードまたは属性ノード (エレメント名の名前マスク) を満たします。
- `NCName ':' '*'` は、名前の接頭部として `NCName` をもつすべてのエレメント・ノードを満たします。
- `QName` は、指定した名前をもつすべてのノードを満たします。

注

`NameTest` の形式の `NodeTest` では、ノードは選択した軸上の基本タイプ、つまり、属性軸上の属性タイプおよび子軸上の子タイプであると想定されます。したがって、`NameTest` が使用できるのは、子ノードおよび属性ノードを選択する場合のみです。コメント・ノードまたは処理命令ノードの選択には使用できません。さらに、このパターンは、ネーム・スペース・ノード以外のあらゆる種類のノードの選択に使用できますが、軸指定子「ネーム・スペース」が許可されていないため、ネーム・スペース・ノードの選択には使用できません。

パターン例は以下のとおりです。

- `chapter | appendix` は、すべての章エレメントおよび付録エレメントを表します。
- `table` はすべての表エレメントを表します。
- `*` はすべてのエレメントを表します (これは `child::*` の省略形です)。
- `ulist/item` は、`ulist` 親をもつすべての項目エレメントを表します。
- `appendix//subsection` は上位の付録をもつすべてのサブセクションを表します。
- `/` はルート・ノードのみを含む `singleton` セットを表します。
- `comment()` はすべてのコメント・ノードを表します。
- `processing-instruction()` はすべての処理命令を表します。
- `attribute::*` (または `@*`) はすべての属性ノードを表します。

以下は、ロケーター・エレメントの構文です。

```

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 および QName は、XML 命名の推奨で定義されたとおりです。

NCName

コロンを含まない XML 名

QName

NCName を先行させることができる NCName で、先行する NCName の後にはコロンが続きます。例： NCName:NCName

テキスト・フィールドおよび文書属性の制限

以下に、テキスト・フィールドおよび文書属性の制限を示します。

- 索引でのフィールドの最大数: 32767
- 1 つの文書での、STRING タイプの 1 つの属性の値の最大数: 1024
- STRING タイプの属性の最大数: 253
- STRING 属性値の文字数は 128 に切り捨てられる
- DATE タイプおよび NUMBER タイプの属性の最大数: 32766
- DATE または NUMBER の属性値の文字数は 128 に切り捨てられる
- NUMBER 属性の場合は、倍精度の浮動小数点数がナンバーとして受け入れられる
- 1 つの文書での、DATE タイプまたは NUMBER タイプの 1 つの属性に指定できる値の最大数: 無制限

HTML 文書モデルに組み込むことのできるタグを以下に示します。

- <A>
- <ADDRESS>
- <AU>
- <AUTHOR>
- <H1>
- <H2>, <H3>, <H4>, <H5>
- <H6>
- <TITLE>

その他のタグを含む <HEAD> および <BODY> などのタグは、HTML 文書モデルでテキスト・フィールドとして指定することはできません。

Outside-In タグ属性値

Outside-In 文書プロパティのタグ・タイプに関連するタグ属性が取ることのできる値を以下に示します。

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

Outside-In 開始および終了タグのサブタイプに関連するタグ属性が取ることのできる値を以下に示します。

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

表にはすべての文書プロパティ、および INSO フィルターが認識するすべてのタグ・サブタイプが含まれているので注意してください。2つのサブタイプ例外があります。SCCCA_DOCUMENTPROPERTY および SCCCA_BOOKMARK。

付録 H. テキスト検索エンジン

DB2 Net Search Extender には、以下のテキスト検索エンジン情報があります。

- トークン化
- ストップワード
- 構成

トークン化

索引作成時に、Net Search Extender は次の方法でテキストをトークンに分割して、文書テキストを処理します。

ワード

すべての英数字 ("a".."z,"A".."Z", "0".."9") が、フルテキスト索引を作成するために使用されます。区切り文字は、ブランク文字および下記の文認識セクションで説明する文字です。改行およびブランク文字などの制御文字は、次のように解釈されません。行の中間にある制御文字 (0x20 より小) は、ブランク文字と見なされません。改行 (0x0A) の前後にあるブランク文字と制御文字は無視されます。1 バイト文字の前後の改行はブランク文字と見なされ、同じ文字の 2 バイト文字は常にその 1 バイト文字と同じものと見なされます。同じ文字の大文字と小文字、たとえば、"A" と "a" は、検索時に特に指定がない場合には同じ文字と見なされ、検索時に正確なマッチングが要求された場合には異なる文字と見なされます。

文

Net Search Extender は、ブランク文字が後に続く「.」「!」「?」、および行末の日本語と中国語の終止符を文の末尾として認識します。

段落

段落の認識は、文書フォーマットによって異なります。プレーン・テキスト・フォーマットでは、2 つの連続する改行文字 (復帰文字が介在する可能性があります) は、段落の境界として認識されます。HTML では、段落タグ <p> が段落の境界として解釈されます。他の文書フォーマットでは、段落の認識をサポートしません。

ストップワード

ストップワードとは、頻繁に現れ、テキスト検索処理に関連する内容をもたないワードのことです。通常、"and"、"or"、および "in" など、すべての (言語的意味での) 機能語がストップワードと見なされます。

Net Search Extender は、言語リストに対してストップワード処理を行います。この場合、ストップワードの索引は作成されません。したがってストップワードを検索することはできません。ただし、ストップワード処理の結果、より小さく高速なテキスト索引になります。

索引付けされていないストップワードは、検索中に普通の語と同じように処理されるので注意してください。ただし、ストップワードが索引付けされている場合は、検索処理中にその語が無視されます。

ストップワードをサポートする言語

以下の言語でストップワード処理を行うことができます。

AR_AA	アラビア語諸国で話されるアラビア語
CA_ES	スペインで話されるカタロニア語
DA_DK	デンマークで話されるデンマーク語
DE_CH	スイスで話されるドイツ語
DE_DE	ドイツで話されるドイツ語
EL_GR	ギリシャで話されるギリシャ語
EN_GB	英国で話される英語
EN_US	米国で話される英語
ES_ES	スペインで話されるスペイン語
FI_FI	フィンランドで話されるフィンランド語
FR_CA	カナダで話されるフランス語
FR_FR	フランスで話されるフランス語
HE_IL	イスラエルで話されるヘブライ語
IS_IS	アイスランドで話されるアイスランド語
IT_IT	イタリアで話されるイタリア語
IW_IL	イスラエルで話されるヘブライ語
NB_NO	ノルウェーで話されるノルウェー語ブークモール
NL_BE	ベルギーで話されるオランダ語
NN_NO	ノルウェーで話されるノルウェー語ニーノシュク
PT_BR	ブラジルで話されるポルトガル語
PT_PT	ポルトガルで話されるポルトガル語
RU_RU	ロシアで話されるロシア語
SV_SE	スウェーデンで話されるスウェーデン語

構成

Net Search Extender は、さまざまな組み合わせで使用される文字 (たとえば英数字、数字、および特殊文字) を含むワードを検索できます。そのために、Net Search Extender には以下の構成が用意されています。

文字正規化

文字正規化は、二通りの書き方があるワードについて、そのどちらからでも両方を検索できるようにします。たとえば、ドイツ語のワード 'über' は 'ueber' と書くこ

ともできます。正規化により、これら双方のワードを 'über' と 'ueber' のどちらからでも検索できます。また、アクセント記号 (たとえば á) も使用可能です。

ワードの一部としての特殊文字の使用

ワードの一部として特殊文字を使用すると、英数字、特殊文字、および数字の連続が含まれることのある製品名を 1 つのワードとして検索対象にできます。たとえば、'/' 特殊文字を使用可能にすることによって、AS/400[®] や OS/390[®] プラットフォームを、'AS' と '400' または 'OS' と '390' としてではなく 1 つのワードとして検索対象にできます。

両方の構成タイプに、スイッチを使用できます。スイッチをカスタマイズするには、索引を作成する前に .ini ファイル・テンプレートを変更します。

.ini ファイル・テンプレートは sqllib/db2ext/cteixcfg.ini に保管されています。CREATE INDEX コマンドを使用してこのテンプレート・ファイルを変更することも可能ですが、その場合以下の部分のみを変更することをお勧めします。

```
UnicodeNormalization      - for character normalization
AccentNormalization       - for character normalization
UmlautNormalization       - for character normalization

AdditionalAlphanumCharacters - for using specific characters as part of
                             a word
```

.ini ファイルにリストされているパラメーターを変更することのみ許可されています。

上記の変更を行った後は、すべての新しい索引が新規の構成で処理されることに注意してください。

スイッチをアクティブにするには、行頭の ";" を削除します。詳細については cteixcfg.ini ファイルを参照してください。

付録 I. テキスト検索エンジン理由コード

- 0 操作が正常に実行されました - エラーは発生しませんでした。
- 1 関数に無効ハンドルが渡されました。
- 2 関数が十分なメモリーを割り振れませんでした。
- 3 アクセス制限またはセキュリティ制限のため、関数が実行されませんでした。
- 4 このバージョンのテキスト検索エンジン・ランタイムでは、この操作はサポートされていません。
- 5 この操作は現在、使用可能になっていません。
- 6 アプリケーションがテキスト検索エンジン・プロトコルに違反しました。正しくない順番でテキスト検索エンジン関数を呼び出しました。
- 7 予期しないエラーが発生しました。エラーをサービス担当者にお知らせください。
- 8 無効な言語が指定されました。
- 9 指定した言語は有効ですが、テキスト検索エンジン・ランタイムでサポートされていません。
- 10 無効な CCSID が指定されました。
- 11 指定した CCSID は有効ですが、テキスト検索エンジン・ランタイムでサポートされていません。
- 12 無効な文書 ID が指定されました。
- 13 指定した文書フォーマットは有効ですが、テキスト検索エンジン・ランタイムでサポートされていません。
- 14 無効な文書フォーマットが指定されました。
- 15 ファイル入出力時のアクセス制限のため、操作が成功しませんでした。
- 16 ファイル入出力時の読み取りエラーのため、操作が成功しませんでした。
- 17 ファイル入力時の読み取りエラーのため、操作が成功しませんでした。
- 18 ファイル出力時の書き込みエラーのため、操作が成功しませんでした。
- 19 ファイル入出力時のシーク・エラーのため、操作が成功しませんでした。
- 20 ファイル入出力時の tell エラーのため、操作が成功しませんでした。

- 21 ファイル入出力時のクローズ・エラーのため、操作が成功しませんでした。
- 22 名前変更操作時のエラーのため、操作が成功しませんでした。
- 23 除去操作時のエラーのため、操作が成功しませんでした。
- 24 mkdir 操作時のエラーのため、操作が成功しませんでした。
- 25 1 つ以上の関数引き数が無効値をもっています (例：予期しない NULL ポインターまたは無効な列挙型値)。
- 26 指定したディレクトリーが存在しません。
- 27 予期しないテキスト検索エンジン・エラーが発生しました。詳細については、エラー情報オブジェクトのテキスト検索エンジン・エラー・コードを参照してください。
- 28 予期しない COS エラーが発生しました。このエラーを報告してください。
- 29 空の文書を更新しようとしてしました。
- 30 この操作では、指定した引き数はサポートされていません。
- 31 日付属性パーサーが、日付属性の解析中に無効値を検出しました。
- 32 数値属性パーサーが、数値属性の解析中に無効値を検出しました。
- 33 属性名が無効です。長過ぎる可能性があります。
- 35 この番号は将来のため予約されています。
- 36 入力文書に、属性の長さ制限を超える属性 (DATE、NUMBER、または STRING) が含まれています。属性テキストがその制限まで切り捨てられました。
- 38 ユーザーが設定した警告しきい値に到達しました。この結果、このエラーが生成されました。
- 39 入力文書に索引を作成できませんでした。含まれているネストされたフィールドが多過ぎます。
- 40 この索引で、1 つの属性タイプに対する異なる属性数の制限を超えました。
- 46 イテレーターのリストが空または削除されているため、イテレーターが有効ではありません (または有効でなくなりました)。
- 47 渡された種類のハンドルでは、この関数はサポートされていません。照会結果イテレーターでないリスト・イテレーターで `itlQueryResultEntryObtainData` を使用した場合などに、このエラーが発生します。
- 48 指定した言語およびリソース・パスでストップワード・ファイルが検出されない場合に、この警告が出されます。
- 49 ストップワード・ファイルにストップワードが含まれていない場合に、この警告が出ます。
- 50 ストップワード・ファイルに無効データが含まれている場合に、この警告が出ます。

- 100 指定した名前またはディレクトリー (あるいはその両方) の索引が存在しないため、索引をオープンできませんでした。
- 101 指定した索引名は無効な索引名です。
- 102 指定した索引ディレクトリーは、無効なディレクトリー名です。
- 103 テキスト索引エンジンが、索引構造または索引ファイル・セットの破壊されたことを検知したため、操作を実行できません。
- 104 指定した名前およびディレクトリーの索引がすでに存在するため、指定した索引を作成できません。
- 109 この索引では、ロールバック操作を実行してからでなければ、その他の操作を実行することはできません。
- 110 索引構成ファイルには、エラー・コンテキストで指定されているような必須セクションは含まれていません。
- 111 索引構成ファイルには、エラー・コンテキストで指定されているような必須オプションは含まれていません。
- 112 索引構成ファイルは、エラー・コンテキストで指定されているような必須オプションに無効データを含んでいます。
- 113 索引構成ファイルは、テキスト検索エンジンのバージョンと一致しません。
- 200 指定した文書モデル名は無効なモデル名です。
- 201 指定した文書モデル・フィールド名は無効なフィールド名です。
- 202 指定した文書モデルは定義されていません。
- 203 指定した文書モデルはすでに存在するため、再定義することはできません。
- 204 索引に追加された文書モデルが、多過ぎるか大き過ぎます。
- 205 文書モデルに含まれるエレメントが多過ぎます。
- 206 文書モデル・エレメントにこのタイプのエレメントには許可されていないパラメーター (XML 属性) が含まれています。
- 207 文書モデル・エレメントにこのタイプのパラメーターには許可されていないパラメーター値 (XML 属性) が含まれています。
- 208 文書モデル・エレメントに、「名前」などの、必要なパラメーター (XML 属性) が含まれていません。
- 209 文書モデルが XML と認識されないか、予期しない XML エレメントで始まっています。
- 210 指定した XPath (ロケーター値) に予期しないトークンが含まれています。
- 211 指定した XPath (ロケーター値) に予期しない軸指定子 (2 つのコロンに続く名前) が含まれています。
- 212 指定した XPath (ロケーター値) に予期しないノード・テストが含まれています。

理由コード

- 213 文書モデル・ディレクトリー・ファイル (拡張子 .mdx) が破壊されています。
- 214 文書モデル索引ファイル (拡張子 .mox) が破壊されています。
- 215 特定の文書属性にマップされ、さらに別の文書属性を含む XML エレメントが文書に含まれています。内部属性は無視されます。
- 216 指定したパラメーター値が、GPP タグまたは HTML タグとして長過ぎます。
- 217 文書モデルに、重複したフィールド定義が含まれています。
- 218 文書モデルに、重複した属性定義が含まれています。
- 300 テキスト索引エンジンが、文書名マッピングに使用される索引ファイルで壊れた部分を検出したため、操作を実行できません。
- 301 テキスト索引エンジンが無効な文書番号を検出したため、操作を実行できません。
- 302 テキスト索引エンジンが無効な文書 ID を検出したため、操作を実行できません。
- 303 テキスト索引エンジンが文書 ID の索引入力を検出しなかったため、操作を実行できません。
- 304 テキスト索引エンジンが文書番号の索引入力を検出しなかったため、操作を実行できません。
- 305 テキスト索引エンジンが、使用された文書番号でオーバーフローを検出したため、操作を実行できません。
- 306 アプリケーションが索引の作成を試みた文書 ID が、文書リストにすでに表示されています。テキスト検索エンジンは、1 つの索引作成順序 (つまり、更新がコミットされる以前の) での重複した文書 ID をサポートしていません。
- 340 用語の強さが無効です。
- 341 関係番号が無効です。in でなければなりません。
- 342 関係タイプが無効です。API に記述されたどれかの定義を使用してください。
- 343 句 (用語) が長過ぎます。
- 344 読み取り中に予期しないファイル終わりが検出されました。
- 345 索引/シソーラス・ファイルの読み取り中にバージョンの矛盾が検出されました。
- 346 シソーラス・バッファーでのオーバーフロー。
- 347 ファイルまたはディレクトリーの名前が無効です。名前が長過ぎる可能性があります。
- 348 検索で辞書の用語 (句) が検出されませんでした。あるいは、定義ファイルの項目に必須用語が含まれていません。
- 349 定義ファイルが空です。

- 350 入力パラメーターで指定されたシソーラス辞書または定義ファイルが存在しません。
- 351 定義ファイルでの構文エラー。
- 352 リレーションシップが誤って指定されました。
- 352 リレーションシップ番号が範囲外です。
- 360 無効な単一文字マスキングが使用されました。
- 361 無効な複数文字マスキングが使用されました。
- 362 演算子 `arity` が、照会で指定されたオペランド数よりも小さい値です。
- 363 `ItlEnOperator` 列挙型で定義された範囲外の演算子値
- 364 列挙型範囲外のランク公式の値。
- 365 近接セグメントを定義する数値が範囲外です。
- 366 照会は準備中であり、再定義またはリセットできません。
- 367 以前の検索結果として与えられた有効範囲が空の結果を示します。
- 368 最初のフィールド名を設定する前にフィールド名の追加を要求している無効な呼び出し。
- 369 索引内容との無効な比較を要求する無効な検索フラグは無視されません。大文字小文字を区別しないで作成された索引に対して、大文字小文字を区別した比較を行うよう要求した場合などに、この理由コードがエラー情報に表示されます。
- 370 タイ語または DBCS 言語では、ストリングのマスキングはサポートされていません。
- 371 有効な照会入力がありません。たとえば、検索項目が使用可能です。
- 372 無効な比較演算が要求されました。
- 373 無効な比較演算が要求されました。
- 374 空の索引に対して検索索引ハンドルが要求されました。
- 375 演算子と、要求された演算子モードとの組み合わせはサポートされていません。
- 380 検索結果は不完全です。しきい値のため、検索が中断されました。
- 381 索引検索によって、照会にストップワードが含まれていることが判明しました。
- 401 テキスト索引エンジンが、フィールド/属性の名前マッピングに使用される索引ファイルに壊れた部分を検出したため、操作を実行できません。
- 402 テキスト索引エンジンが無効なフィールド名または属性名を検出したため、操作を実行できません。
- 403 指定したフィールド名または属性名は定義されていないため、操作を実行できません。

- 404 この索引で、属性タイプのいずれかに対する異なる属性の制限、あるいは異なるフィールドに対する制限を超えました。
- 500 文書/データに無効な文字シーケンスが含まれています (UTF8 または UTF16 または DBCS のソース内で)。
- 501 コード・ページ・コンバーターにエラーが発生しました。
- 502 文書/データに不完全な文字シーケンスが含まれています (UTF8 または UTF16 または DBCS のソース内で)。
- 503 コード・ページ・コンバーターに無効な記述子があります。
- 600 XML 文書に非同期エンティティーが含まれています。たとえば、引用符で囲まれていない XML 属性値。
- 602 無効な文字参照 (or など)。
- 603 無効なバイナリー・エンティティー参照。
- 604 XML パーサー Expat を作成できませんでした。
- 605 タグの属性名は重複しない名前であればなりません。
- 607 XML パーサーは無効な外部エンティティー参照を検出しました。
- 608 文書には、< または > などの欠落した、誤ったトークンが含まれています。
- 609 XML 文書は囲みタグをもっている必要があります。この囲み終了タグの後では、テキストは許可されていません。
- 610 この位置に処理命令を使用することはできません。たとえば、最初の処理命令を `prolog <?xml .. ?>` にすることはできません。
- 611 エレメントは、開始タグ、内容、終了タグのシーケンスです。「<s>text /s>」などの場合、終了タグが誤っているため、このエラーが発生します。
- 612 XML パーサーでメモリー割り振りが失敗しました。
- 614 無効なパラメーター・エンティティー参照です。
- 615 不完全な文字です。2 バイト UTF8 文字の最初のバイトだけの可能性があります。
- 616 再帰的エンティティー参照です。
- 617 XML 構文エラー。たとえば、囲み開始タグおよび終了タグの外にテキストがある場合など。
- 618 すべての開始タグには組み合わせの終了タグが必要です。
- 619 データ・セクションが閉じられていません。
- 620 トークンが閉じられていません。たとえば、文書内の最後のトークンの後にテキストがある場合など。
- 621 解決できない文書がエンティティー内にあります。
- 622 予期しないエラーです。
- 631 メタ・タグ内のフィールド情報または属性情報を解析できませんで

した。タグは <meta name="abc" content="xyz"> のフォーマットにする必要があります。メタ・タグの属性名または内容が誤っている可能性があります。

- 632 エンティティーを文字に変換できませんでした。
- 650 異なるフィールド定義が同じ開始タグで始まっています。
- 651 開始タグに別の開始タグが含まれているため、タグがあいまいになっています。
- 652 フィールドおよび属性に同じ開始タグを使用する場合、それらには同じ終了タグを使用するか、両方ともに終了タグを使用しないようにする必要があります。
- 653 文書が終了していますが、フィールドが閉じられていません。
- 654 構造化フォーマットに対して文書モデルが指定されていません。文書は、フィールド情報または属性情報のないプレーン・テキストとして構文解析されます。
- 670 「Outside In」(TM) ライブラリーが必要ですが、検出できなかったため、操作を実行できませんでした。
- 671 「Outside In」(TM) ライブラリーから必要なプロシーチャーをロードできなかったため、操作を実行できませんでした。ライブラリーは有効期限が切れているか、破壊されている可能性があります。
- 672 「Outside In」で文書在处理中にエラーが発生しました。

理由コード

付録 J. トラブルシューティング

DB2 Net Search Extender は、該当するコマンドを使用しない障害のトレースおよび DB2 オブジェクトのドロップについての以下の参照情報を提供します。

障害のトレース

IBM 担当員にエラーを知らせる必要がある場合には、エラーの原因を検出するために使用するファイルに情報を書き込めるようにするために、トレースをオンにするように依頼されることがあります。

トレース機能をオンにするとシステム・パフォーマンスに影響が出るため、トレース機能は、IBM サポート担当者またはお客様の技術サポート担当者から依頼があった場合にだけ使用してください。

トレースをオンにするには、次の DB2 機能を使用します。

```
db2trc on
```

詳細については、「*DB2 UDB コマンド・リファレンス (SC88-9140)*」を参照してください。

Net Search Extender にユニークな情報を受け取るには、次のようにコンポーネントのマスクに 96 を指定します。

```
db2trc on -m *.*.96.*.*
```

重大エラーの場合には、db2diag.log を参照すると役に立ちます。

正しい Net Search Extender コマンドを使用しない DB2 オブジェクトのドロップ

DB2 Net Search Extender は、正しいコマンドを使用せずに DB2 オブジェクトをドロップする場合のために、以下の参照情報を提供します。

表のドロップ

1 つ以上のテキスト索引を持つ表をドロップする前に、各テキスト索引に対して以下のコマンドを発行することが必要です。

```
db2text drop index <index_name> for text
```

誤って、索引をドロップする前に表をドロップしてしまうと、管理表やテキスト索引ファイルなど一部の索引が残ってしまいます。

これらのファイルを除去するには、表がすでに存在していないとしても、db2text drop index コマンドを使用して索引をドロップします。

データベースのドロップ

1 つ以上のテキスト索引を持つデータベースをドロップする前に、各テキスト索引に対して以下のコマンドを発行します。

```
db2text drop index <index_name> for text
```

このコマンドを使用しない場合は、`index_directory` と `index_work_directory` にあるすべての索引ファイルを手動で削除する必要があります。

ドロップしたデータベースに属する索引が自動更新中に作成された場合には、スケジューラー・ファイル `ctedem.dat` の編集が必要になることに注意してください。

その処理を行うには、以下を実行します。

UNIX の場合:

```
db2text stop force  
cd ~/sql1lib/db2ext
```

または Windows の場合:

```
db2text stop force  
cd <db2_install_path>%sql1lib%\<db2_instance_name>%db2ext
```

ディレクトリー内のファイル `ctedem.dat` を開き、ドロップしたデータベースを参照するすべての項目を削除します。

付録 K. データ・リンク・メッセージ

データ・リンクからのエラーで Net Search Extender エラーになることはなく、索引作成処理が終了することはありません。必ず、イベント・ログをチェックし、索引が作成されていない文書に注意してください。必要に応じてそれらの文書の索引が再作成されるように、手動で処置を行う必要があります。

表 18. データ・リンク警告メッセージ

番号	データ・リンク・メッセージ
01H90=	CTEDL - 戻り Blob 値を設定する際のエラー。
01H91=	CTEDL - データ・リンク入出力操作がタイムアウトになりました。
01H92=	CTEDL - 文字エンコード方式がサポートされていません。
01H93=	CTEDL - サポートされていないデータ・リンク方式。
01H94=	CTEDL - データ・タイプ BLOB のインスタンスを作成する際のエラー。
01H95=	CTEDL - プロキシ情報を設定する際のエラー。
01H96=	CTEDL - UNC スキーマは Windows 上でのみ有効ですが、OS を判別できませんでした。
01H97=	CTEDL - UNC スキーマは Windows 上でのみ有効です。
01H98=	CTEDL - DFS スキーマは AIX 上でのみ有効です。
01H99=	CTEDL - DFS スキーマは AIX 上でのみ有効ですが、OS を判別できませんでした。
01H01=	CTEDL - URL スtringのポート番号を判別する際のエラー。
01H02=	CTEDL - 不明なデータ・リンク方式が検出されました。
01H03=	CTEDL - 接続を確立できませんでした。
01H00=	CTEDL - データ・リンク UDF を実行する際のエラー。
01H80=	CTEDL - データ・リンク・ファイル "{0}" が見つかりませんでした。
01H81=	CTEDL - "{0}" に対する無許可のデータ・リンク・アクセス。
01H82=	CTEDL - "{0}" について、予期しないファイル終わりまたはストリームの終わりに到達しました。
01H83=	CTEDL - データ・リンク・ファイル "{0}" が読み取り不能です。
01H85=	CTEDL - データ・リンク URL 方式 "{0}" にはファイル名が必要です。
01H86=	CTEDL - データ・リンク・ファイル・サーバー "{0}" への接続が確立していません。
01H60=	CTEDL - 誤った HTTP 要求 - 誤った形式のデータ・リンク URL 構文。
01H61=	CTEDL - 無許可のデータ・リンク要求 - ユーザー認証が必要です。
01H62=	CTEDL - データ・リンク・アクセスには支払いが必要です。
01H63=	CTEDL - データ・リンク URL に対する禁止アクセス。
01H64=	CTEDL - データ・リンク・サーバーでファイルが見つかりませんでした。
01H65=	CTEDL - データ・リンク・リソースでは要求された方式が許可されていません。
01H66=	CTEDL - 要求を受け入れることができません。

表 18. データ・リンク警告メッセージ (続き)

01H67=	CTEDL - プロキシ認証が必要です。
01H68=	CTEDL - クライアント要求のタイムアウト。
01H69=	CTEDL - データ・リンク・リソースの現在の状態との競合。
01H10=	CTEDL - サーバーでデータ・リンク・リソースを使用できなくなっています。
01H11=	CTEDL - 要求を受け入れるには、内容の長さを指定する必要があります。
01H12=	CTEDL - ヘッダー・フィールドに指定された前提条件が false であると判断されました。
01H13=	CTEDL - 要求されたデータ・リンク・エンティティが大き過ぎます。
01H14=	CTEDL - 要求されたデータ・リンク URL が長過ぎます。
01H15=	CTEDL - サポートされていない MIME タイプ。
01H16=	CTEDL - 範囲要求を満たすことができません。
01H17=	CTEDL - 予想と異なりました。
01H18=	CTEDL - データ・リンク URL への要求に内容の長さに関する情報がありません。
01H20=	CTEDL - HTTP 応答が無効です。
01H70=	CTEDL - 内部データ・リンク・サーバー・エラー。
01H71=	CTEDL - データ・リンク・サーバーによって機能がサポートされていません。
01H72=	CTEDL - 誤ったゲートウェイ。
01H73=	CTEDL - サービスが使用できません - DL サーバーが一時的に過負荷になっているか、保守されています。
01H74=	CTEDL - ゲートウェイのタイムアウト。
01H75=	CTEDL - HTTP バージョンがサポートされていません。
01H30=	CTEDL - ソケット接続を確立してください - 基礎を成すプロトコルのエラー。
01H31=	CTEDL - データ・リンク・サーバー {0} への経路を確立できませんでした。
01H32=	CTEDL - ソケットをリモート・アドレス {0} に接続できませんでした。
01H33=	CTEDL - ソケットをローカル・アドレスに結合できませんでした。
01H34=	CTEDL - データ・リンク・サーバー {0} の IP アドレスを判別できませんでした。
01H35=	CTEDL - 不明なサービス例外 - MIME タイプはサポートされていません。
01H36=	CTEDL - 誤った形式の URL '{0}' - サポートされたプロトコルまたは DL URL の構文解析を行うことができませんでした。

付録 L. シソーラスでサポートされている CCSID

以下の CCSID がシソーラスでサポートされています。

CCSID

819	Latin 1
850	PC データ Latin 1
874	タイ語
932	結合された日本語
943	結合された日本語
950	結合された中国語 (繁体字)
954	日本語
970	結合された韓国語
1208	UTF 8
1250	Latin 2
1252	Latin 1
1253	チェコ語
1254	トルコ語
1255	ヘブライ語
1256	アラビア語
1258	ベトナム語
1363	結合された韓国語
1381	結合された中国語 (簡体字)
1383	中国語 (簡体字)、結合された SBCS/DBCS
1386	中国語 (簡体字)、結合された SBCS/DBCS
5039	日本語 (結合された SBCS/DBCS)

シソーラス定義ファイルをコンパイルするには、139 ページの『DB2EXTTH (ユーティリティ)』を参照してください。

付録 M. シソーラス・ツールが戻すメッセージ

ADM_MSG_INVALID_CCSID

説明: 無効な CCSID が指定されました。
要求されたコード・ページはサポートされていません。

ITL_THES_MSG_DEFFILE_MISSING

説明: パラメーター・エラー *file name*。シソーラス定義ファイルが存在しません。

ITL_THES_MSG_NONAME_ERROR

説明: パラメーター・エラー。シソーラス定義ファイル名が指定されていません。

ITL_THES_MSG_PATHLEN_ERROR

説明: パラメーター・エラー *file name*。シソーラス定義ファイル・パスが長過ぎます。パスの長さは、オペレーティング・システムのディレクトリー名でサポートされている最大長を超えてはなりません。

ITL_THES_MSG_NAMELEN_ERROR

説明: パラメーター・エラー *file name*。シソーラス定義ファイル名が長過ぎます。

ITL_THES_MSG_NO_TARGET_DIR_ERROR

説明: パラメーター・エラー。ターゲット・ディレクトリーが指定されていません。

ITL_THES_MSG_UNEXPECTED_ERROR

説明: 予期しない内部エラー。

ITL_THES_MSG_PARAMETER_ERROR

説明: 内部パラメーター・エラー。

ITL_THES_MSG_FILE_OPEN_ERROR

説明: ファイル *file name* をオープンできませんでした。

ITL_THES_MSG_FILE_REACHED_END

説明: シソーラス定義ファイルに予期しないファイル終わりがあります。

定義ファイルにエラーがあります。

ITL_THES_MSG_FILE_READ_ERROR

説明: ファイル *file name* を読み取れませんでした。

ITL_THES_MSG_FILE_WRITE_ERROR

説明: ファイル *file name* に書き込むことができませんでした。

ITL_THES_MSG_FILE_ACCESS_ERROR

説明: ファイル *file name* にアクセスできませんでした。

ITL_THES_MSG_FILE_REMOVE_ERROR

説明: ファイル *file name* を除去できませんでした。

ITL_THES_MSG_FILE_RENAME_ERROR

説明: ファイル *file name 1* の名前を *file name 2* に変更できませんでした。

ITL_THES_MSG_FILE_CLOSE_ERROR

説明: ファイル *file name* をクローズできませんでした。

ITL_THES_MSG_FILE_EOF_ERROR

説明: *file name* に予期しないファイル終わりがあります。

定義ファイルにエラーがあります。

ITL_THES_MSG_MEMORY_ERROR

説明: メモリー・エラー。

ITL_THES_MSG_BUFFER_OVERFLOW

説明: バッファー・オーバーフロー。

ITL_THES_MSG_LOCKING_ERROR

説明: 辞書 *file name* をロックできませんでした。

シソーラス・ツールが戻すメッセージ

ITL_THES_MSG_LOCKED

説明: シソーラス辞書 *dictionary name* は使用中です。

ITL_THES_MSG_OUTFILE_EXIST

説明: 出力ファイル *file name* はすでに存在しています。

ITL_THES_MSG_DICT_INTEGRITY_ERROR

説明: 辞書 *dictionary name* の整合性が失われています。

シソーラス辞書ファイルが壊れています。

ITL_THES_MSG_DICT_VERSION_ERROR

説明: 辞書 *dictionary name* バージョン・エラー。

シソーラス辞書が互換性のない古いバージョンによって作成されています。

ITL_THES_MSG_DICT_NOT_EXIST

説明: シソーラス辞書 *dictionary name* が存在しません。

ITL_THES_MSG_DICT_EXIST

説明: シソーラス辞書 *dictionary name* がすでに存在しています。

上書きできません。

ITL_THES_MSG_NORMALIZE_ERROR

説明: 用語を正規化する際のエラー。

シソーラス定義ファイルにエラーがあります。

ITL_THES_MSG_INTERNAL_ERROR

説明: 内部エラー

ITL_THES_MSG_INPUT_ERROR

説明: シソーラス定義ファイル *file name* の行 *line number* にエラーがあります。

ITL_THES_MSG_ERROR_IN_FILE

説明: ファイル *file name* のエラー。

ITL_THES_MSG_IE_EMPTY

説明: シソーラス定義ファイル *file name* が空です。

ITL_THES_MSG_IE_BLOCK_START

説明: ファイル *file name* の行 *line number* で、ブロック開始行が見つかりませんでした。

ITL_THES_MSG_IE_REL_SYNTAX

説明: *file name* の行 *line number* でリレーションシップが誤って指定されています。

ITL_THES_MSG_IE_USER_DEF

説明: *file name* の行 *line number* でリレーションシップが誤って指定されています。

ITL_THES_MSG_IE_USER_DEF_DOMAIN

説明: *file name* の行 *line number* でリレーションシップ番号が範囲外になっています。

ITL_THES_MSG_IE_NO_TERM

説明: *file name* の行 *line number* で用語が定義されていません。

ITL_THES_MSG_IE_TERM_LEN

説明: シソーラス用語が 64 文字より長くなっています。

ITL_THES_MSG_IE_STRENGTH_SYNTAX

説明: 強さの値が誤って指定されています。

構文: 用語の後に、強さ 20 の場合は [:20] と入力してください。

ITL_THES_MSG_IE_STRENGTH_DOMAIN

説明: 強さが範囲外です。

有効値は 1 ~ 100 です; デフォルトは 100 です。

付録 N. Windows システム・エラー

以下に、Windows システム・エラーのリストを示します。

システム・エラー

- | | |
|-----|-----------------------------------|
| 1 | ファンクションが間違っています。 |
| 2 | 指定されたファイルが見つかりません。 |
| 3 | 指定されたパスが見つかりません。 |
| 4 | ファイルを開くことができません。 |
| 5 | アクセスが拒否されました。 |
| 6 | ハンドルが無効です。 |
| 8 | このコマンドを実行するのに十分な記憶域がありません。 |
| 14 | この操作を完了するのに十分な記憶域がありません。 |
| 15 | 指定されたドライブが見つかりません。 |
| 29 | 指定されたデバイスに書き込めません。 |
| 30 | 指定されたデバイスから読み取れません。 |
| 32 | プロセスはファイルにアクセスできません。別のプロセスが使用中です。 |
| 36 | 開かれている共有ファイルが多すぎます。 |
| 38 | ファイルの終わりです。 |
| 39 | ディスクがいっぱいです。 |
| 80 | ファイルがあります。 |
| 82 | ディレクトリまたはファイルを作成できません。 |
| 100 | ほかのシステム セマフォを作成できません。 |
| 101 | 排他セマフォは、ほかのプロセスが所有しています。 |
| 102 | セマフォが設定されています。閉じることができません。 |
| 103 | セマフォを再設定できません。 |
| 104 | 割り込み時間には排他セマフォを要求できません。 |
| 105 | このセマフォの以前の所有権は終了しました。 |
| 110 | 指定されたデバイスまたはファイルを開けません。 |
| 111 | ファイル名が長すぎます。 |
| 112 | ディスクに十分な空きスペースがありません。 |
| 121 | セマフォがタイムアウトしました。 |
| 126 | 指定されたモジュールが見つかりません。 |

- 127 指定されたプロシージャが見つかりません。
- 147 このコマンドを処理するにはリソースが足りません。
- 155 ほかのスレッドを作成できません。
- 161 指定されたパスは無効です。
- 164 このシステムでは、これ以上のスレッドを作成できません。
- 170 要求されたリソースは使用中です。
- 183 既に存在するファイルを作成することはできません。
- 187 指定されたシステム セマフォ名が見つかりません。
- 206 ファイル名または拡張子が長すぎます。
- 267 ディレクトリ名が無効です。
- 288 呼び出し側が所有していない mutex を解放しようとしています。
- 298 1 つのセマフォに対するポストが多すぎます。
- 998 メモリ ロケーションへのアクセスが無効です。
- 1051 別の実行中のサービスが依存しているサービスにストップ コントロールが送信されました。
- 1052 要求された制御はこのサービスに対して無効です。
- 1053 そのサービスは指定時間内に開始要求または制御要求に応答しませんでした。
- 1054 サービスに対してスレッドを作成できませんでした。
- 1055 サービス データベースはロックされています。
- 1056 サービス インスタンスは既に実行されています。
- 1057 アカウント名が無効か、または存在しません。
- 1058 指定されたサービスは無効であるか、または有効なデバイスが関連付けられていないため、開始できません。
- 1059 循環するサービス依存関係が指定されました。
- 1060 指定されたサービスはインストールされたサービスとして存在しません。
- 1061 そのサービスは現時点でコントロール メッセージを受け付けることができません。
- 1062 サービスが開始していません。
- 1063 サービス プロセスをサービス コントローラに接続できませんでした。
- 1064 制御要求を処理しているときに、サービスで例外が発生しました。
- 1066 そのサービスからサービス固有エラー コードが返されました。
- 1067 プロセスを途中で強制終了しました。
- 1068 依存関係サービスまたはグループを起動できませんでした。
- 1069 ログオンに失敗したため、サービスを開始できませんでした。

- 1070** サービスは開始後に開始待ち状態でハングしました。
- 1071** 指定されたサービス データベース ロックは無効です。
- 1072** 指定されたサービスは削除の対象としてマークされています。
- 1073** 指定されたサービスは既に開始されています。
- 1078** その名前は、サービス名またはサービス表示名として既に使われています。
- 1079** このサービスに対して指定されたアカウントは、同じプロセス内で実行中のほかのサービスに対して指定されたアカウントと異なります。
- 1082** このサービスに構成されたりカバリー・プログラムが何もありません。
- 1154** このアプリケーションの実行に必要なライブラリ ファイルの 1 つが壊れています。
- 1219** 入力内容が、既存の資格情報のセットと一致しませんでした。
- 1242** そのサービスは既に登録されています。
- 1243** 指定されたサービスはありません。
- 1244** ユーザーが認証されていないため、要求された操作は実行されませんでした。
- 1245** ユーザーがネットワークにログオンしていないため、要求された操作は実行されませんでした。指定されたサービスはありません。
- 1392** ファイルまたはディレクトリが壊れているため、読み取ることができません。
- 1455** ページング ファイルが小さすぎるため、この操作を完了できません。
- 1793** ユーザーのアカウントは有効期限が切れています。

付録 O. 特記事項

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-0032
東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。 IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Canada Limited
Office of the Lab Director
1150 Eglinton Ave. East
North York, Ontario
M3C 1H7
CANADA

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができませんが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性がありますが、その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生した創作物には、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. _年を入れる_. All rights reserved.

商標

以下は、IBM Corporation の商標です。

AIX	DB2 Universal Database
DB2	IBM DRDA
DB2 Extenders	z/OS
Informix	OS/400
OS/390	AS/400
zSeries	

以下は、それぞれ各社の商標または登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

MMX および Pentium は、Intel Corporation の米国およびその他の国における商標です。

UNIX は、The Open Group の米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

用語集

この用語集は、本書で使用されている用語および省略語を定義しています。搜している用語がない場合は、索引を参照するか、*Dictionary of Computing*, New York: McGraw-Hill, 1994 を参照してください。

[ア行]

アクセス関数 (access function). 列に保管されたテキストのデータ・タイプを DB2 Net Search Extender によって処理できるタイプに変換するユーザー提供の関数。

一致 (match). テキスト文書での検索項目の出現。

エスケープ文字 (escape character). 以降の文字がマスク文字 であると解釈されないことを示す文字。

[カ行]

カウント (count). 指定された関係の検索項目を拡張するために使用される、シソーラスでの項目のレベルの数 (深さ) を指定するために使用するキーワード。

拡張 (expand). シソーラスから派生した追加の用語を検索項目に追加する処置。

カタログ・ビュー (catalog view). 管理目的のために DB2 Net Search Extender Text によって作成されたシステム表のビュー。カタログ・ビューには、DB2 Net Search Extender Text が使用できるように使用可能化された表および列に関する情報が含まれている。

関数 (function). アクセス関数 (*access function*) を参照。

検索引き数 (search argument). 検索を実施する際に指定された条件で、1 つまたは複数の検索項目、および検索パラメーターから構成される。

コード・ページ (code page). すべてのコード・ポイントに GRAPHIC 文字および制御機能の意味を割り当てること。例: 文字および意味を 8 ビット・コードでの 256 コード・ポイントに割り当てること。

更新頻度 (update frequency). 日、時間、および分で表現され、索引付けが可能になる前に、索引付けのためにログ表 にリストする必要がある文書名の最少数による、テキスト索引の更新頻度。

コマンド行プロセッサ (command line processor). 次の機能を持つ db2text と呼ばれるプログラム。

DB2 Net Search Extender コマンドを入力できるようにする

コマンドを処理する

結果を表示する

[サ行]

索引 (index). テキストから重要な用語を抜き出して、テキスト索引 に保管すること。

索引特性 (index characteristics). 次の内容を決定するテキスト索引 のプロパティ。

索引が更新される頻度

最初の索引更新が発生する時期

周期的索引付け (periodic indexing). 事前に定義された日、時間、および分で表現された時間間隔、および索引付けが可能になる前に、索引付けのためにログ表 にリストする必要がある文書名の最少数による索引付け。

使用可能化 (enable). データベースを DB2 Net Search Extender が使用できるように準備すること。

使用不可 (disable). 使用可能化処理中に作成された項目を除去することによって、データベースを DB2 Net Search Extender Text のために使用可能化される前の状態にリストアすること。

スコア (Score). ある文書が、他の検出された文書と比較して検索基準を満たす度合いを示す、0 と 1 の間のタイプ DOUBLE の絶対値。この値は、文書内で検出された一致の数を、文書のサイズとの関係で示している。

[タ行]

テキスト索引 (text index). テキスト文書から抽出された重要な用語の集合。各用語は、抽出元の文書と関連付けられている。文書自体ではなく、索引での検索によって検索時間が大幅に短縮される。

テキスト文書 (text document). DB2 表に保管されるタイプ CHAR、GRAPHIC、VARGRAPHIC、LONG VARGRAPHIC、DBCLOB、VARCHAR、LONG VARCHAR、または CLOB データ・タイプのテキスト。

テキスト列 (text column). テキスト文書を含む列。

トリガー (trigger). 文書がテキスト列との間で追加、変更、または削除されるたびに、ログ表に索引付けするために必要な文書に関する情報を、自動的に追加するメカニズム。

トレース (tracing). 後でエラーの原因を調べる際に使用できる情報をファイルに保管する処置。

[ハ行]

ハイブリッド検索 (hybrid search). ブール検索 (*Boolean search*) とフリー・テキスト検索 (*free-text search*) との結合されたもの。

ブール検索 (Boolean search). 1 つまたは複数の検索項目がブール演算子を使用して結合されている検索。

ファジー検索 (fuzzy search). スペルが検索項目のスペルに類似している語を検出できる検索。

フォーマット (format). ASCII または HTML などの文書のタイプ。

フリー・テキスト検索 (free-text search). 検索項目がフリー・フォーム・テキスト (検索される対象の自然言語で記述される句または文) として表現される検索。

文書 (document). テキスト文書 (*text document*) を参照。

文書モデル (document model). 文書に含まれるセクションに関する、文書の構造の定義。文書モデルは、索引付け時に、DB2 Net Search Extender に文書内のセクションを認識させる。文書モデルは、セクションを識別するマークアップ・タグをリストする。そのセクションに対して照会で使用するための記述的なセクション名をタグごとに指定できる。1 つまたは複数の文書モデルを文書モデル・ファイルに指定することができる。

[マ行]

マスク文字 (masking character). 検索項目の先頭、中間、および末尾のオプションの文字を表現するために使用される文字。マスク文字は、通常、正確な索引で用語のバリエーションを検出するために使用される。

[ヤ行]

ユーザー定義関数 (UDF) (user-defined function (UDF)). DB2 によって提供された SQL 関数とは異なり、DB2 のユーザーが作成した SQL 関数。

ユーザー定義タイプ (UDT) (user-defined type (UDT)). LONG VARCHAR などの DB2 によって提供されたデータ・タイプとは異なり、DB2 のユーザーが作成したデータ・タイプ。

[ラ行]

リトリブ (retrieve). DB2 Net Search Extender の検索関数の 1 つの検索引き数を使用してテキスト文書を検索すること。

ログ表 (log table). どのテキスト文書が索引付けされるかに関する情報を含む、DB2 Net Search Extender によって作成された表。使用可能にされたテキスト列の文書が追加、変更、または削除されるたびに、トリガーを使用してこの情報がログ表に保管される。

[ワ行]

ワイルドカード文字 (wildcard character). マスク文字 (*masking character*) を参照。

C

CCSID. コード化文字セット ID (Coded Character Set Identifier)。

D

DBCS. 2 バイト文字セット (Double-byte character set)。

S

SBCS. 1 バイト文字セット (Single-byte character set)。

U

UDF. ユーザー定義関数 (User-defined function)。

UDT. ユーザー定義タイプ (User-defined type)。

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

移行 191

一致

検索結果内 74

NUMBEROFMATCHES 関数 157

イベント・ビュー 205

インスタンス・サービス 9, 27, 97

インストール 13

インストール, データ・リンク jar ファイルの 38

インストール, パーティション化された DB2 サーバーの 14

インストール検査 16

インフォメーション・カタログ 199

エスケープ文字

使用法 77

[カ行]

開始, DB2 Net Search Extender の 100
概念

インスタンス・サービス 9

主要な 3

ストアード・プロシージャ検索の使用 6

追加概念 8

ビュー 9

表値関数の使用 7

列変換関数 8

SQL スカラー検索関数の使用 5

外部保管データ 9

概要 3

拡張, 検索項目の

参照: シソーラス

カタログ・ビュー 33

環境, クライアント/サーバー 11

関係, シソーラスでの 90

関数

概要 155

検索関数 73

検索引き数の指定 75

参照情報 155

ストアード・プロシージャ 168

説明 73

関数 (続き)

データ・タイプの変換用 37

テキストの検索 74

CONTAINS 156

HIGHLIGHT 163

NUMBEROFMATCHES 157

SCORE 158

SQL 表値 159

管理

インスタンス所有者のコマンド・サマリ 97

キャッシュのアクティブ化 71

キャッシュの非アクティブ化 71

索引イベントのクリア 48, 71

索引状況の表示 72

索引のバックアップとリストア 50

障害のトレース 257

データベース管理者コマンドのサマリ 103

テキスト索引状況の表示 49

テキスト索引設定の変更 48, 67

テキスト索引の更新 46, 70

テキスト索引の作成 35, 55

テキスト索引のドロップ 49, 69

テキスト索引の保守 46, 66

テキスト表所有者のコマンド・サマリ 111

ロッキング・サービスの使用 27

DB2 Net Search Extender の開始 27, 52, 100

DB2 Net Search Extender の停止 27, 52, 101

DB2 コントロール・センター 51

「キャッシュのアクティブ化」ダイアログ 71

「キャッシュ表」パネル 62

「キャッシュ表の非アクティブ化」ダイアログ 71

クライアント/サーバー環境 11

計画 23

言語, サポートされている 215

検索, テキストの

概要 74

検出された一致の数の入手 74

検出された文書のスコアの入手 75

構文 148

照会を行う 74

ストアード・プロシージャの使用 80

表値関数の使用 81

HIGHLIGHT 関数の使用 82

検索関数

ストアード・プロシージャ 168

CONTAINS 156

HIGHLIGHT 163

NUMBEROFMATCHES 157

SCORE 158

SQL 表値 159

検索項目の拡張

参照: シソーラス

検索の例, 複数の列の 84

検索引き数

同じ段落内での用語の検索 78

同じ文または同じ段落内での用語の検索 78

構文 148

固定順序での用語検索 77

シソーラス検索 78

指定 75

数値属性検索 79

説明 147

属性名 153

任意の順序での用語検索 75

ブール演算子 150

ファジー検索 76, 151

フリー・テキスト検索 79

文書のセクション内での用語の検索 78

マスク文字の使用 77

用語の一部の検索 77

ワイルドカード文字の使用 77

NOT を使用した検索 78

search-primary 演算子 150

& と | を使用して検索する 76

検索引き数キーワード

COUNT 153

EXPAND 153

EXPANSION LIMIT 150

FUZZY FORM OF 151

PRECISE FORM OF 151

RESULT LIMIT 150

SECTION 150

STEMMED FORM OF 151

STOP SEARCH AFTER number DOCUMENTS(S) 150

TERM OF 153

THESAURUS 153

更新サービス 30

更新頻度 46

構造化文書

概要 87

検索構文 150

構造化文書 (続き)

セクション・サポートを可能にする

171

デフォルトの文書モデル 174

例 78

コマンド

ACTIVATE CACHE 112

ALTER INDEX 114

CLEAR EVENTS 118

CONTROL 98

COPYRIGHT 146

CREATE INDEX 120

DB2EXTDL 108

DB2EXTHL 109

DB2EXTTH 139

db2text 97, 103, 111

DEACTIVATE CACHE 135

DISABLE DATABASE 106

DROP INDEX 137

ENABLE DATABASE 104

HELP 145

START 100

STOP 101

UPDATE INDEX 141

コマンド・サマリー

インスタンス所有者のための 97

データベース管理者用の 103

テキスト表所有者のための 111

[サ行]

サーバー

開始 100

障害のトレース 257

停止 101

索引

概要 3

関係 90

キャッシュのアクティブ化 71

キャッシュの非アクティブ化 71

計画 23

更新頻度 46

構造化文書の使用 87

サイズの計算 23

索引イベントのクリア 48, 71

索引状況の表示 72

テキスト索引状況の表示 49

テキスト索引設定の変更 48, 67

テキスト索引の更新 46, 70

テキスト索引のドロップ 49, 69

テキスト索引の保守 46, 66, 89

バックアップとリストア 50

DB2 コントロール・センター 51

「索引イベント」ダイアログ 71

索引更新イベント

記録 36

索引更新イベント (続き)

削除 48

「索引状況」ダイアログ 72

「索引の更新」ダイアログ 70

索引の再作成 46

「索引のドロップ (Drop Index)」ダイアログ 69

索引のバックアップとリストア 50

「索引の変更」ダイアログ 67

「サマリー」パネル 66

サンプル関数

実行 80

システム要件 13

シソーラス

概念 89

構造 89

コンパイル 91

作成 91

サポートされている CCSID 261

シソーラス定義構文 187

定義ファイル 91

メッセージ 263

シソーラス検索

構文 153

例 78

THESAURUS キーワード 153

シソーラス・コンパイル・ユーティリティ
ー 139

主キー・タイプ 133

出現、検索語の 157

主要な概念 3

主要な機能 10

主要な用語 3

使用、大量のメモリーの 195

AIX の場合 195

HP-UX 197

Linux 197

Sun Solaris の場合 196

Windows の場合 196

使用、ワードの一部としての特殊文字の
247

障害の検出 257

ストアード・プロシージャ

概要 40

テキスト索引のアクティブ化 43

テキスト索引の更新 42

テキスト索引の非アクティブ化 43

ビューに基づくテキスト索引 43

DB2 レプリケーションを使用して、
ニックネームのテキスト索引を作成
する 38

ストアード・プロシージャ関数

検索 80

構文 168

ストップワード 245

スペース所要量、索引の 23

[タ行]

「ターゲット」パネル 57

追加概念 8

データ

外部保管 9

データベース

索引のバックアップとリストア 50

データベースを使用可能にする 32,
53

データベースを使用不可にする 33,
53

停止、DB2 Net Search Extender の 101

ディスク・スペース、索引の 23

ディレクトリー名とファイル名 16

テキスト検索エンジン

構成 246

ストップワード 245

ストップワードをサポートする言語
246

トークン化 245

理由コード 249

テキスト特性

フォーマット 23

CCSID 23

「テキスト・プロパティ」パネル 59

デフォルトの文書モデル 174

「特性の更新」パネル 60

トラブルシューティング 257

トリガー

作成 35

説明 5

トレース、障害の 257

[ナ行]

「名前」パネル 55

入門 19

認識

ストップワード 245

段落 245

文 245

文字正規化 246

ワード 245

ワードの一部としての特殊文字の使用
247

[ハ行]

パネルおよびダイアログ

「キャッシュのアクティブ化」ダイア
ログ 71

「キャッシュ表」パネル 62

「キャッシュ表の非アクティブ化」ダ
イアログ 71

「索引イベント」ダイアログ 71

パネルおよびダイアログ (続き)
「索引状況」ダイアログ 72
「索引の更新」ダイアログ 70
「索引のドロップ (Drop Index)」ダイアログ 69
「索引の変更」ダイアログ 67
「サマリー」パネル 66
「ターゲット」パネル 57
「テキスト・プロパティ」パネル 59
「特性の更新」パネル 60
「名前」パネル 55
パフォーマンスに関する考慮事項
検索に関する 84
索引作成のための 45
汎用パーサー (GPP) 文書
制限 242
文書フォーマット 23
文書モデルの定義 175
文書モデルの文書タイプ定義 239
ビュー
イベント・ビュー 205
概要 9
作成されたビュー 33
レプリケーション・キャプチャー表 206
ログ表ビュー 206
db2ext.dbdefaults 33, 199
db2ext.indexconfiguration 33, 203
db2ext.proxyinformation 33, 201
db2ext.textindexes 33, 49, 202
db2ext.textindexformats 33, 204
表値関数
ビューに基づくテキスト索引 43
レプリケーションを使用してニックネームのテキスト索引を作成する 38
表値検索関数
検索 81
構文 159
HIGHLIGHT 関数の使用 82
HIGHLIGHT 構文 163
表スペース 35, 57
ブール演算子
検索構文 150
NOT 78
& (AND) および | (OR) 76
ファジー検索、例 76
フォーマット、テキスト文書の 23
サポート対象のリスト 23
説明 23
深さの指定、シソーラスの項目の 153
フラット ASCII、文書フォーマット 23
付録
移行 191
インフォメーション・カタログ 199
サポートされている言語 215

付録 (続き)
シソーラスでサポートされている
CCSID 261
ストップワード 245
大量のメモリの使用 195
データ・リンク・メッセージ 259
テキスト検索エンジン 245
テキスト検索エンジン構成 246
テキスト検索エンジン理由コード 249
トークン化 245
トラブルシューティング 257
文書モデル 239
メッセージ、シソーラス・ツールが戻す 263
CCSID 209
Net Search Extender メッセージ 217
Windows システム・エラー 265
文書
構造 171
索引付け 3
サポートされるフォーマット 23
データ・タイプの変換 37
フォーマット、説明 23
CCSID 23
文書タイプ 23
文書データ・タイプ
サポートされないデータ・タイプの変換 37
バイナリー・データ・タイプ 36
DATALINK データ・タイプ 38
文書モデル
概要 87
検索構文の SECTION キーワード 150
検索構文の属性名 153
構造化文書の記述 171
制限 242
説明 171
テキスト・フィールド 173
デフォルト 174
文書属性 173
文書タイプ定義 239
変更 171
例 171
文書モデルの参照情報 239
変更、データ・リンクの結果サイズの 38, 53

[マ行]

マスク文字、検索語での 77
メモリー量 195
文字正規化 246

[ヤ行]

ユーザーの役割
データベース管理者 26
テキスト表所有者 26
DB2 インスタンス所有者 25
ユーザー・シナリオ
ストアード・プロシージャ検索の例 21
SQL スカラー検索の例 19
SQL 表値関数の例 22

[ラ行]

列変換関数 8
レプリケーション・キャプチャー表 126, 206
ログ表
作成 35
説明 5
ログ表ビュー 206
ロッキング・サービス
使用法 27
表示 29
CONTROL コマンド 98

[ワ行]

ワイルドカード文字、検索語での 77

A

ACTIVATE CACHE コマンド
構文 112
使用法 43
AIX でのインストール 15
ALTER INDEX コマンド
構文 114
使用法 48
ASCII、文書フォーマット 23

C

CCSID
シソーラスでサポートされている 261
文書コード・ページ 23
リスト 209
CLEAR EVENTS コマンド
構文 118
使用法 48
COMMITCOUNT
キーワード 129
パフォーマンスに関する考慮事項 45
CONTAINS 関数
構文 156

CONTAINS 関数 (続き)

例 74

CONTROL コマンド

構文 98

使用法 29

COPYRIGHT コマンド

構文 146

COUNT キーワード 153

CREATE INDEX コマンド

構文 120

使用法 35

D

Datalink Manager

エラー・メッセージ 259

データ・リンクの結果サイズの変更
38, 108

DATALINK データ・タイプ 38

jar ファイルのインストール 38

DB2 Net Search Extender の概要 3

DB2 コントロール・センター

ウィザードを使用する 55

管理 51

「キャッシュのアクティブ化」ダイア
ログ 71

「キャッシュ表」パネル 62

「キャッシュ表の非アクティブ化」ダ
イアログ 71

「索引イベント」ダイアログ 71

「索引状況」ダイアログ 72

「索引の更新」ダイアログ 70

「索引のドロップ (Drop Index)」ダイ
アログ 69

「索引の変更」ダイアログ 67

「サマリー」パネル 66

「ターゲット」パネル 57

データベースを使用可能にする 53

データベースを使用不可にする 53

テキスト索引の作成 55

テキスト索引の保守 66

「テキスト・プロパティ」パネル
59

「特性の更新」パネル 60

「名前」パネル 55

DB2 Net Search Extender の開始およ
び停止 52

DB2EXTDL コマンド

構文 108

使用法 139

DB2EXTHL コマンド

構文 109

db2ext.dbdefaults ビュー 199

db2ext.indexconfiguration ビュー 203

db2ext.proxyinformation ビュー 201

db2ext.textindexformats ビュー 204

DB2TX、コマンド行プロセッサ

構文 97, 103, 111

使用法 31

DEACTIVATE CACHE コマンド

構文 135

使用法 43

DISABLE データベース コマンド

構文 106

使用法 33, 53

DROP INDEX コマンド

構文 137

使用法 49

E

ENABLE データベース コマンド

構文 104

使用法 32, 53

EXPAND キーワード 153

EXPANSION LIMIT キーワード 150

F

FUZZY FORM OF キーワード 151

H

HELP コマンド

構文 145

HIGHLIGHT 関数

構文 163

例 164

CLOB サイズの変更 109

TEXTSEARCH 関数の使用 163

HTML 文書

構造化文書 171

制限 242

デフォルトの文書モデル 174

文書フォーマット 23

文書モデルの定義 177

文書モデルの文書タイプ定義 239

I

INSO フォーマット

参照: Outside-In フィルター・ソフト
ウェア 16

N

Net Search Extender

インスタンス・サービス 27

開始および停止 27, 52

キャッシュのアクティブ化 71

Net Search Extender (続き)

キャッシュの非アクティブ化 71

更新サービス 30

索引イベントのクリア 48, 71

索引状況の表示 72

ストアード・プロシージャ検索用キ
ャッシュの作成 40

データベースを使用可能にする 32,
53

データベースを使用不可にする 33,
53

テキスト索引状況の表示 49

テキスト索引設定の変更 48, 67

テキスト索引の更新 46, 70

テキスト索引の作成 35, 55

テキスト索引のドロップ 49, 69

テキスト索引の保守 46, 66

バックアップとリストア 50

メッセージ 217

DB2 コントロール・センターの使用
51

DB2 レプリケーションを使用して、
ニックネームのテキスト索引を作成
する 38

Net Search Extender インフォメーショ
ン・カタログ

ビューを参照 199

NUMBEROFMATCHES 関数

構文 157

例 74

O

OR ブール演算子 76

Outside-In フィルター・ソフトウェア
概要 24

構造化文書 171

タグ属性 243

デフォルトの文書モデル 174

文書フォーマット 23

文書モデルの定義 184

ライブラリーのインストール 16

P

PRECISE FORM OF キーワード 151

R

RESULT LIMIT キーワード 150

S

SCORE 関数

構文 158

SCORE 関数 (続き) 1 (OR) 演算子、検索引き数の
例 75 使用方法 76
search-primary 演算子 150
START コマンド
構文 100
使用法 27, 52
STEMMED FORM OF キーワード 151
STOP SEARCH AFTER number
DOCUMENTS(S) キーワード 150
STOP コマンド
構文 101
使用法 27, 52

T

TERM OF キーワード 153
TEXTSEARCH 関数
構文 159
例 161
HIGHLIGHT 関数の使用 159

U

UNIX でのインストール 14
UNIX でのインストール検査 17
UPDATE INDEX コマンド
更新頻度 46
構文 141
使用法 47
ストアド・プロシージャ用 42
RECREATE オプション 46

W

Windows システム・エラー 265
Windows でのインストール 15
Windows でのインストール検査 16

X

XML 文書
構造化文書 171
制限 242
デフォルトの文書モデル 174
文書フォーマット 23
文書モデルの定義 179
文書モデルの文書タイプ定義 239
XPath 式のセマンティクス 240
XPath 式のセマンティクス 240

[特殊文字]

& (AND) 演算子、検索引き数の
使用方法 76



部品番号: CT2XLJA

Printed in Japan

SH88-8546-02



日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木3-2-12

(1P) P/N: CT2XLJA



Spine information:



IBM® DB2 Universal
Database

DB2 Net Search Extender
管理およびユーザーズ・ガイド

バージョン 8.2