

DB2 Universal Database



**Net Search Extender
管理およびユーザース・ガイド**

バージョン 8.1

DB2 Universal Database



**Net Search Extender
管理およびユーザース・ガイド**

バージョン 8.1

ご注意!

本書、および本書がサポートする製品をご使用になる前に、253 ページの『付録 N. 特記事項』にある一般的な情報を必ずお読みください。

本書は、IBM Universal Database Net Search Extender バージョン 8.1、(プログラム番号:5765-F38) に適用されます。また、改訂版等で特に断りのない限り、これ以降のすべてのリリースおよびモディフィケーション・レベルにも適用されます。

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)

原 典： SH12-6740-00
DB2 Universal Database
Net Search Extender
Administration and User's Guide
Version 8.1

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 2002.10

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, 2002. All rights reserved.

© Copyright IBM Japan 2002

目次

本書について	vii	4. サンプル・データをロードする	18
本書の対象読者	vii	5. テキスト索引を同期させる	19
本書の使用法	vii	6. テキスト索引で検索する	19
構文図の読み方	viii	キャッシュを使用したストアード・プロシ ジャー検索の場合の簡単な例	19
関連情報	ix	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	20
第 1 部 ユーザーズ・ガイド	1	2. 索引を同期させ、キャッシュを活動化す る	20
第 1 章 概要と概念	3	3. TEXTSEARCH ストアード・プロシ ジャーで検索する	20
主要な概念	3	SQL 表値関数の場合の簡単な例	20
SQL スカラー検索関数の使用	6	第 4 章 計画	23
ストアード・プロシジャー検索の使用	7	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
検索での SQL 表値関数の使用	8	文書フォーマットおよびサポートされるコー ド・ページ	23
追加概念	9	ユーザーの役割	24
列変換関数	9	DB2 インスタンス所有者	24
インスタンス・サービス	9	データベース管理者	25
外部保管データ	9	テキスト表所有者	25
管理ビュー	10	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	25
主要な機能	10	第 5 章 Net Search Extender インスタ ンス・サービス	27
DB2 クライアント / サーバー環境内の DB2 Net Search Extender	11	DB2 Net Search Extender の開始および停止	27
第 2 章 インストール	13	ロッキング・サービス	27
システム要件	13	ロッキング・サービスの使用	28
AIX および Sun-Solaris システムでのインス トール	14	ロック・スナップショットの表示	29
AIX 用のステップ 1: 製品コンポーネン トのインストール	14	更新サービス	30
Sun-Solaris 用のステップ 1: 製品コンポー ネントのインストール	14	第 6 章 テキスト索引の作成と保守	31
AIX および Solaris 用のステップ 2: DB2 インスタンスの更新	15	db2text コマンドの紹介	31
Windows でのインストール	15	データベースの使用可能化	32
ディレクトリーとファイル名	16	データベースのディセープリング	33
インストール検査	16	テキスト索引の作成	34
第 3 章 ユーザー・シナリオ	17	バイナリー・データ・タイプのテキスト索 引の作成	36
SQL スカラー検索関数の場合の簡単な例	17		
1. データベースをテキスト検索可能にする	18		
2. 表を作成する	18		
3. フルテキスト索引を作成する	18		

サポートされないデータ・タイプのテキスト索引の作成	36	ストアード・プロシージャ検索を使用するテキスト検索	85
DATALINK データ・タイプのテキスト索引の作成	37	SQL 表値関数を使用する検索	86
ストアード・プロシージャ検索で使用するテキスト索引の作成	38	パフォーマンスに関する考慮事項	87
ビューに基づくテキスト索引	41	第 9 章 構造化文書を使用する作業	89
パフォーマンスに関する考慮事項	43	第 10 章 検索項目を拡張するためのシソーラスの使用	91
テキスト索引の保守	43	シソーラスの構造	91
テキスト索引の更新と再編成	43	定義済みシソーラス関係	92
テキスト索引の変更	45	ユーザー独自の関係の定義	93
索引イベントのクリア	46	シソーラスの作成およびコンパイル	94
テキスト索引をドロップする	46	シソーラス定義ファイルの作成	94
テキスト索引状況を表示する	47	定義ファイルのシソーラス辞書へのコンパイル	95
索引のバックアップとリストア	47	第 2 部 参照情報	97
第 7 章 DB2 コントロール・センターの使用	49	第 11 章 インスタンス所有者用の管理コマンド	99
DB2 Net Search Extender インスタンス・サービスの開始および停止	50	CONTROL	100
データベースの使用可能化	51	START	102
テキスト索引の管理	52	STOP	103
テキスト索引の作成	53	第 12 章 データベース管理者用の管理コマンド	105
テキスト索引の保守	68	ENABLE DATABASE	106
第 8 章 検索	77	DISABLE DATABASE	108
SQL スカラー検索関数を使用するテキストの検索	78	DB2EXTDL (ユーティリティ)	110
照会を行う	78	第 13 章 テキスト表所有者用の管理コマンド	111
検索と見つかった一致数の戻し	79	ACTIVATE CACHE	113
検索と検出したテキスト文書のスコアの戻し	79	ALTER INDEX	115
SQL 検索引き数の指定	80	CLEAR EVENTS	119
任意のシーケンスでの用語検索	80	CREATE INDEX	121
ブール演算子 AND と OR を使用する検索用語の一部の検索 (文字のマスキング)	81	DEACTIVATE CACHE	133
マスク文字をすでに含んでいる用語の検索	82	DROP INDEX	135
固定シーケンスでの用語検索	82	DB2EXTTH (ユーティリティ)	137
任意のシーケンスでの用語検索	82	UPDATE INDEX	139
同じ文または同じ段落内での用語検索	82	HELP	142
構造化文書のセクション内での用語検索	83	COPYRIGHT	144
ブール演算子 NOT を使用する検索	83	第 14 章 検索引き数の構文	145
ファジー検索	83	検索引き数	146
シソーラス検索	84		
数値属性検索	84		
フリー・テキスト検索	84		
その他の検索構文の例	85		

第 15 章 SQL スカラー検索関数および SQL 表値関数.	153	ログビュー	194
検索関数のサマリー	153	付録 D. サポートされている CCSID	195
CONTAINS	154	SBCS	195
NUMBEROFMATCHES	155	DBCS	198
SCORE	156	UNICODE	199
DB2EXT.TEXTSEARCH	157	付録 E. サポートされている言語	201
第 16 章 ストアード・プロシージャ検索関数	161	付録 F. Net Search Extender メッセージ	205
DB2EXT.TEXTSEARCH (ストアード・プロシ ージャー検索用)	162	通知および警告メッセージ	205
第 17 章 構造化文書のサポート	165	エラー・メッセージ	206
文書モデル	165	付録 G. 文書モデルの参照情報	225
デフォルトの文書モデル	165	文書モデル用の DTD	225
構造化プレーン・テキストの場合の文書モデ ルの定義	166	ロケーター (XPath) 式のセマンティクス	226
エレメント・パラメーター	167	テキスト・フィールドおよび文書属性の制限	228
GPP 文書を索引付けするとどうなるか	168	付録 H. テキスト検索エンジン	231
HTML 文書の場合の文書モデルの定義	169	トークン化	231
エレメント・パラメーター	170	ワード	231
XML 文書の場合の文書モデルの定義	171	文	231
エレメント・パラメーター	172	段落	231
XML 文書を索引付けするとどうなるか	175	ストップワード	231
第 18 章 シソーラス・サポート	177	ストップワードをサポートする言語	232
第 3 部 付録	181	付録 I. テキスト検索エンジン理由コード	233
付録 A. マイグレーション	183	付録 J. トラブルシューティング	241
付録 B. 大量のメモリーの使用	185	障害のトレース	241
AIX	185	付録 K. データ・リンク・メッセージ	243
Windows 32 ビット	185	付録 L. シソーラスでサポートされている CCSID	247
Sun Solaris	186	付録 M. シソーラス・ツールが戻すメッセー ジ	249
付録 C. Net Search Extender インフォメ ーション・カタログ	187	付録 N. 特記事項	253
データベース・レベル情報のビュー	187	商標	255
db2ext.proxyinformation 表	189	用語集	257
索引レベル情報のビュー	189	索引	261
db2ext.textindexes ビュー	190	IBM と連絡をとる	267
db2ext.indexconfiguration ビュー	192	製品情報	267
db2ext.textindexformats ビュー	192		
テキスト索引の表ビュー	193		
イベント・ビュー	193		

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、DB2 Net Search Extender 提供の 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 の関数および API を使用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 関数に関する参照情報およびメッセージやコードなどの診断情報を示します。

DB2 Net Search Extender の概念およびタスクについての知識はあるが、特定の DB2 Net Search Extender の関数、コマンド、メッセージ、またはコー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。

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。

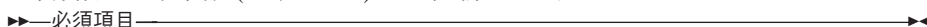
- 構文図は、左から右、上から下に、線に沿って読みます。記号 \blacktriangleright は、ステートメントの始まりを示します。

記号 \longrightarrow 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。

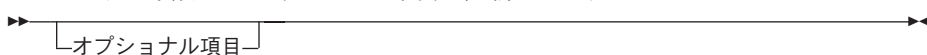
記号 \blacktriangleright は、ステートメントが前の線からの続きであることを示します。

記号 \blacktriangleleft 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す。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**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 (GC88-9141-00)。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、マイグレーション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*」SC88-9168。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*」SC88-9146。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。
- 「*Global Text 9 Programming Guide*」GC09-2846。この資料は、Global Text 9 の概念、文書の索引付けおよび文書内の情報の検索などの情報検索の目的に API 関数を使用する方法について説明しています。

IBM DB2 Universal Database Enterprise-Extended 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。これらの資料は、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。

本書について

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・ファミリーに含まれます。

DB2 Net Search Extender バージョン 8.1 は、DB2 Text Information Extender バージョン 7.2、および Net Search Extender バージョン 7.2 に代わり、SQL 照会を使用して 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検索するための、高速で多様なインテリジェント方式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 ユニバーサル・データベース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、一意的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。

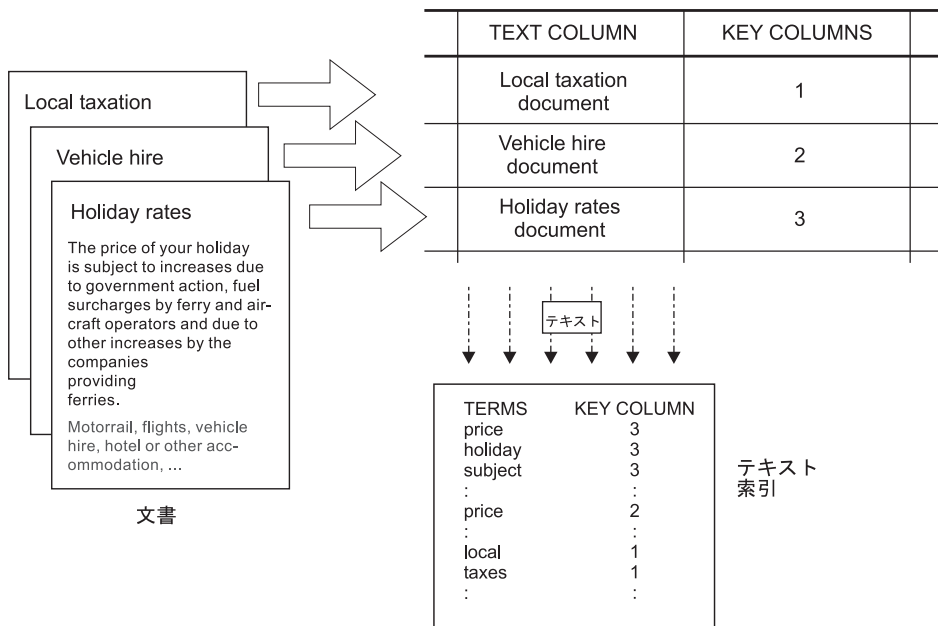


図1. テキスト索引の作成

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・テーブル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・テーブルが存在します。ログ・テーブル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。

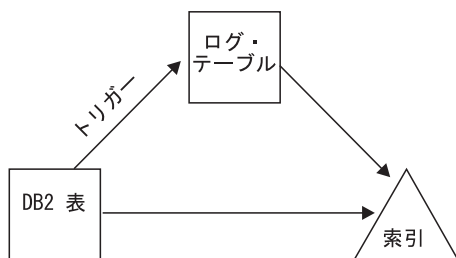


図2. インクリメンタル更新処理

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、索引は読み取りアクセス専用としてロックされます。検索操作ではこのことはわかりません。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などの、特定のプロパティをもっています。必要に応じて、一部のプロパティを変更することができます。これは索引の**変更**と呼ばれます。

そのようなプロパティとして、表の列に関する特定の ORDER BY 句に従って、テキスト索引の事前ソートを行うべきかどうかというプロパティがあります。そのような場合は、初期更新で指定した順序でテキスト文書の索引が作成され、索引結果はこの順序で戻されます。たとえば、本の価格に従って本の要約を事前ソートするように指定する場合があります。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。事前ソートされた索引がないと、すべての本を索引し、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、それ以後の更新が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

主要な概念

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・テキスト検索関数
- ストアド・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

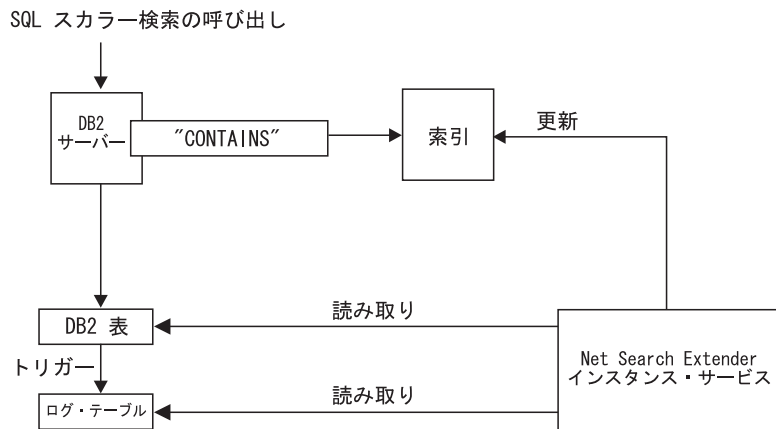


図3. 検索での SQL スカラー検索関数の使用

Net Search Extender には、SQL 内でシームレスに統合される 3 つのスカラー検索関数 (CONTAINS、NOOFMATCHES、および SCORE) があります。これらのスカラー検索関数は、SQL 照会での標準 SQL 式を使用するのと同じ場所で使用することができます。典型的な照会は以下のとおりです。

```
SELECT * FROM books WHERE CONTAINS (abstract, "relational databases")  
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 スカラー検索関数は、ほとんどの状況に適合するため、デフォルトの検索方式として使用してください。特に、テキスト検索式が異なる他の条件と結合している場合は、これを使用してください。

DB2 Optimizer 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。

ストアード・プロシージャ検索の使用

ストアード・プロシージャ検索関数は、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、ストアード・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、使用する前に活動化する必要があります。また、対応する非活動化コマンドがあります。

TextSearch ストアード・プロシージャ検索の呼び出し

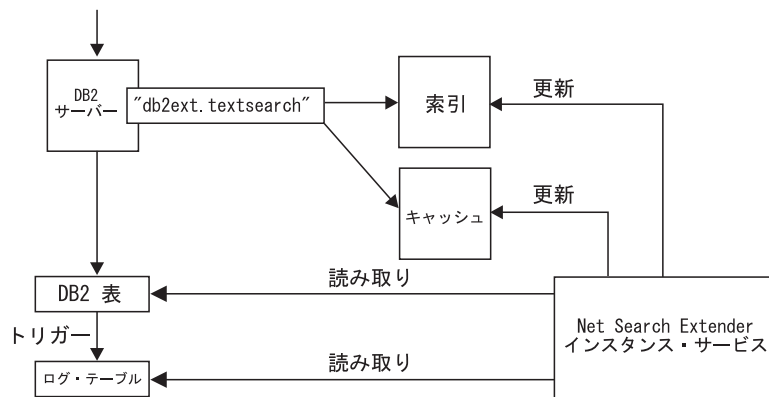


図4. ストアード・プロシージャ検索の使用

ACTIVATE コマンドを使用すると、データは一時キャッシュ（活動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。

検索にストアード・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。

主要な概念

ストアード・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・テーブルに追加するか、RECREATE オプションを使用して作業することができます。

ストアード・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ハイスケーラビリティのアプリケーションに使用してください。このような検索の例として、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があります。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。

- ストアード・プロシージャ検索は、不定の 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す。
- ストアード・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ます。
- ストアード・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。

検索での SQL 表値関数の使用

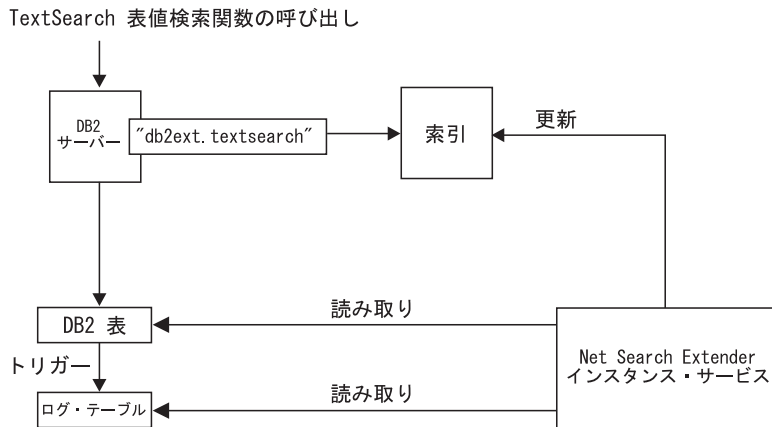


図5. 検索での SQL 表値関数の使用

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。

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。
- 表値関数は事前ソートされたテキスト索引を利用できる。

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または事前ソートされた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たものに変換することができます。これを実行するには、入力として、オリジナルのテキスト文書を読み込むユーザー定義関数 (UDF) を指定します。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す必要があります。

36 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、テキスト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 (自動および手動の双方) 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、CLOBS または VARCHARS などのネイティブ DB2 表内に保管されています。

追加概念

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、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。

ix 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。9 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、187 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。

その他の概念および問題の詳細については、23 ページの『第 4 章 計画』を参照してください。

主要な機能

DB2 Net Search Extender v8.1 の主要な機能は以下のとおりです。

- 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
 - 索引作成用にコマンド行か、または DB2 コントロール・センターを使用した使いやすいインターフェースの選択が可能
 - HTML、XML などのいろいろなテキスト・フォーマット
 - ニックネーム表サポート
 - DB2 Datalink Manager サポート
 - 事前ソートされたテキスト索引のサポート
- 検索

3 つの検索インターフェースが提供されています。SQL スカラー検索関数、ストアード・プロシージャー検索、SQL 表値関数は、以下の検索機能をサポートしています。

 - 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
- 検索引き数を自然言語で表せる特定のテキストを含んでいる文書のフリー・テキスト検索
- 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
- 多数の同時ユーザーによる、大量のテキスト文書に対する高速検索
- 検索結果
 - 検索結果のソート方法を、索引作成時に指定する機能
 - 大量のデータの中から検索して大量の結果リストが予想される場合に、検索結果のサブセットを指定可能
 - 検索項目に対して、ヒット・カウントによる制限を設定可能
 - 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せん。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、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

本章では、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。

インストール後、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 の実行に必要なソフトウェアのバージョンは、下記のとおりです。

- DB2 バージョン 8.1

Sun Solaris の場合は以下のとおりです。

- Solaris 7 (32 ビット) パッチ 106327-10
- Solaris 8 (32 ビット) パッチ 108434-03 および 108528-12
- Solaris 9 (32 ビット)
- Java をサポートするには、以下のパッチも必要です。
 - Solaris 7 "Recommended & Security Patches" + 107226-17 + 107153-01
 - Solaris 8 "Recommended & Security Patches" + 108921-12 + 108940-24

AIX の場合は以下のとおりです。

- AIX® バージョン 4.3.3 (保守レベル 9 以降、32 ビット) ファイル・セット xlc.aix43.rte 5.0.2.x も必要です。このファイル・セットは、製品 CD 上にあります。
- AIX® バージョン 5.1.0 (保守レベル 2 以降、32 ビット)

Windows の場合は以下のとおりです。

- Windows NT® バージョン 4 (Service Pack 6a を適用したもの) 以上
- Windows 2000 (32 ビット)
- Windows XP (32 ビット)
- Windows .NET (32 ビット)

DB2 の資料については、ix ページの『関連情報』を参照してください。

すべてのプラットフォーム上で、DB2 Net Search Extender は 32 ビット・モードで実行されることに注意してください。

インストール

DB2 Net Search Extender を稼働するのに必要な最小ハードウェアは、DB2 Universal Database v8.1 の場合と同じです。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。

AIX および Sun-Solaris システムでのインストール

AIX および Sun-Solaris の場合のインストール・ステップを以下に示します。

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。

AIX 用のステップ 1: 製品コンポーネントのインストール

AIX にインストールするには、次のようにします。

1. ターゲット・マシンに root ユーザーとしてログインします。
2. `installp` 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。システム管理インターフェース・ツール (SMIT) を使用して、指示インターフェースから `installp` を実行することも、`installp` コマンドを入力することもできます。

`smit` を実行するには、次のようにします。

- a. `'smit install_latest'` と入力して、「ソフトウェアのインストールおよびメンテナンス・ツール (The Software Installation and Maintenance Tool)」メニューを表示します。
- b. 「ソフトウェアの入力装置 / ディレクトリー (INPUT device / directory for software)」フィールドにインストール・ファイルのディレクトリーを入力します。

```
/<cdrom>/aix
```

ここで `<cdrom>` は CD-ROM ドライブのパスです。

- c. 「実行 (DO)」ボタンをクリックするか、Enter キーを押します。これによって、インストール・ディレクトリーを確認します。
- d. 「インストールするソフトウェア (SOFTWARE TO INSTALL)」フィールドで、インストールする DB2 Net Search Extender コンポーネントを識別します。
- e. 「実行 (DO)」ボタンをクリックするか、Enter キーを押します。
- f. インストール・パラメーターの確認を求められます。Enter キーを押して、確認します。

Sun-Solaris 用のステップ 1: 製品コンポーネントのインストール

Sun Solaris にインストールするには、次のようにします。

1. ターゲット・マシンに root ユーザーとしてログインします。

2. パッケージをインストールします。パッケージをインストールするには、'pkgadd' コマンドを使用します。

```
pkgadd -d /<cdrom-dir>/solaris/db2nse81.img
```

ここで <cdrom-dir> は CD-ROM ドライブのパスです。選択可能なパッケージのリストが表示されるので、db2nse81 パッケージを選択できます。

AIX および Solaris 用のステップ 2: DB2 インスタンスの更新

DB2 インスタンスを確立または更新するには、次のようにします。

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よって、以下のいずれかのコマンドを入力します。

```
For AIX:      cd /usr/opt/db2_08_01/instance
```

```
For Solaris: cd /opt/IBM/db2/V8.1/instance
```

3. 以下のように指定して db2iupdt を実行します。./db2iupdt <db2instance>。ここで、<db2instance> は、Net Search Extender を使用する必要がある既存の 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

db2icrt 時に、Net Search Extender の新しい DB2 インスタンスが自動的に作成されます。

Windows でのインストール

Windows にインストールするには、次のようにします。

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 インスタンスで、ユーザー ID およびパスワードを入力して、正しいサービスが作成されるように注意してください。
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。
4. この製品をアンインストールするには、コントロール・パネルの「プログラムの追加 / 除去 (Add/Remove)」アイコンを使用します。

注

DB2 インスタンスごとに、NT サービスが作成されます。DB2 サービス用にログオンしたユーザーが、Windows ユーザー名を使用するシステム・アカウントではなく、当該アカウントとして実行中であることを確認してください。

ディレクトリーとファイル名

すべての Net Search Extender コマンドで、ディレクトリーとファイル名は SBCS 文字で指定する必要があります。パス名 (ファイル名を含む) の最大長は 256 バイトまたはそれ以下です。

インストール検査

Net Search Extender インストール検査スクリプトを実行します。

- sql1lib/samples/db2ext に変更します
- nsesample yourdb を呼び出します
- 出力ファイルをチェックします
- db2 connect to yourdb を呼び出します
- db2-tvf search を呼び出します

サンプルが正常に動作すれば、Net Search Extender は正しくインストールされています。

注

マイグレーションについては、183 ページの『付録 A. マイグレーション』を参照してください。

第 3 章 ユーザー・シナリオ

この章では、以下の 2 つの簡単な実例を使用して、Net Search Extender について学習します。

- SQL 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・プロシージャの例。このコマンド行の例では、SQL 例の索引作成コマンドを使用していますが、キャッシュが追加されたために別の索引作成および検索関数も使用可能であることを示しています。

Net Search Extender の使用についての詳細は、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』および 77 ページの『第 8 章 検索』の各章を参照してください。

注

この例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。

(MORE) インストール検査および表値関数。

SQL スカラー検索関数の場合の簡単な例

この DB2 Net Search Extender の例には以下の 6 つのステップがあります。

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

1. 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 コマンドを実行できます。たとえば、

```
db2text "START"
```

により、DB2 Net Search Extender インスタンス・サービスが始まります。

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

このコマンドは、データベースを DB2 Net Search Extender で使用できるように準備します。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。また、データベースの作成直後にこのステップを実行できます。

2. 表を作成する

サンプル表を作成するには、以下のように DB2 コマンドを実行します。

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

このコマンドで author および story の列を含む books と呼ばれる表が作成されます。

3. フルテキスト索引を作成する

フルテキスト索引を作成するには、以下のようにします。

```
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 です。

4. サンプル・データをロードする

表にサンプル・データをロードするには、以下のようにします。

```
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、および story information がロードされます。

5. テキスト索引を同期させる

sample 表に追加されたデータでテキスト索引を更新するには、以下のようにします。

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

6. テキスト索引で検索する

テキスト索引を検索するには、以下のように CONTAINS SQL 検索述部を実行します。

```
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、"\" が使用されています。

この照会では、2000 年以前の、cat に関するすべての本が検索されます。以下の結果表が戻ります。

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。最初の関数は、見つかった文書が検索引き数によって説明されている程度を示すインディケーターを戻し、2 番目の関数は、結果の文書で照会用語が一致した個数を戻します。

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

この DB2 Net Search Extender ストアード・プロシージャ検索の例には以下の 3 つのステップがあります。

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、キャッシュを活動化する。
3. TEXTSEARCH ストアード・プロシージャで検索する。

注

ストアード・プロシージャの例では、前の例のステップが実行されており、データベースがまだ使用可能であると想定しています。

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

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

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

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。

2. 索引を同期させ、キャッシュを活動化する

表に挿入されたデータに基づいて索引を更新するには、以下のようになります。

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

キャッシュを活動化するには、以下のようになります。

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

3. TEXTSEARCH ストアド・プロシージャで検索する

DB2 Net Search Extender TEXTSEARCH ストアド・プロシージャは、すべてのストアド・プロシージャ・コンテキストで使用できます。以下のように、DB2 CALL1 ステートメントで、ストアド・プロシージャを呼び出します。

```
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 つの結果だけが戻されます。この場合、以下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、ディレクトリ `sql1lib\sample\db2ext` 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、前の 2 つの例で作成されたテキスト索引を処理することができます。

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。19 ページの『6. テキスト索引で検索する』を参照してください。

```
db2 "SELECT author, story FROM books b, table (db2ext.textsearch (story,
  \"cat\", 'DB2EXT', 'MYTEXTINDEX
', 0, 2, CAST
  (NULL AS VARCHAR(18))) T where T.primKey = B.isbn
```

注

キャスト (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。

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の設計者、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。

この後の章では、DB2 Net Search Extender に基づくアプリケーションを開発する方法を説明しています。

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。テキスト索引の作成については、31 ページの『第 6 章 テキスト索引の作成と保守』、および 77 ページの『第 8 章 検索』を参照してください。

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、索引付けするデータのサイズとタイプによって異なります。索引用のガイドラインとしては、単一バイト文書の場合、索引付けする文書のサイズの約 0.7 倍のディスク・スペースを確保します。2 バイト文書の場合は、索引付けする文書と同サイズを確保します。

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 ～ 4.0 倍です。

複数の大きな索引がある場合、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。

サポートされる文書フォーマットは、下記のとおりです。

TEXT	プレーン・テキスト (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)

文書フォーマットである HTML、XML、および GPP の場合は、検索は文書の特定部分に制限することができます。89 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は、ユーザー定義関数 (UDF) を作成できます。この関数は索引作成時に指定する必要があり、これを使用して、データを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変換します。

詳細については、121 ページの『CREATE INDEX』を参照してください。

文書は、そのコード・ページ (CCSID) がサポートされていれば、索引付けできます。サポートされるコード・ページのリストは 195 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。詳細については 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。

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

このユーザーは、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、ロックング・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コントロールが可能になります。

- 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 / 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、および DB2TEXT CONTROL。

コマンドは、サーバー上でのみ許可されます。分散 DB2 環境ではサーバーは、どのようなサーバーでもかまいません。また、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。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

この役割を持つユーザーは、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 のためには 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

このユーザーは索引の作成、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コントロール、およびフルテキスト索引の更新のコントロールができません。注意してください。

- 必須 DB2 権限 / 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

コマンドのインプリメンテーションは、DB2 インスタンス所有者のユーザー ID の下で部分的に実行されることに注意してください。したがって、索引の作成 / 変更の前に、必要なファイル・システム・アクセスがインスタンス所有者に認可されている必要があります。

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

索引作成では、コマンド行を使用するか、DB2 コントロール・センター・インターフェースを使用するかを選択できます。

ただし、すべての検索結果を必要としない場合、SQL スカラー検索関数を使用すると、オプティマイザーのサポートを受けることができます。これに対し、ストアード・プロシージャ検索および SQL 表値関数を使用すると、カーソル・サポートしか受けることができません。

注

検索構文は 3 つのオプションすべてに対して同じであることに留意してください。

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、およびロッキング・サービスと更新サービスについて詳しく説明します。

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。

これを実行するには、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
```

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
```

DB2 インスタンスあたりに 1 つの Net Search Extender ロッキング・サービスがなければならぬ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キングを保持します。

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されます。ロッキング・サービスは、マルチユーザー環境で 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 ロッキング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、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、テキスト索引への変更を計算し、その後、ファイルに書き込む必要があるのか、ということに応じて、索引への同時アクセスを制御するいろいろなタイプのロックがあります。

db2text start 時に、ロッキング・サービスは自動的に開始されます。テキスト索引のロックには次の 4 つのタイプがあります。

- **S-lock.** 共用読み取り専用アクセス用。たとえば検索要求
- **U-lock.** 同時読み取りアクセス (たとえば検索要求) に対する索引への変更 (更新) を行なっている間の読み取り / 書き込みアクセス用
- **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キングを保持します。

ロッキング・サービス構成ファイルは、db2ext1m.cfg と呼ばれ、UNIX システムの場合は、<DB2_INSTOWNERHOMEDIR>/sql1lib/db2ext 上に、Windows の場合は、<sql1lib>%db2ext 上に保管されます。

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。詳細については 100 ページの『CONTROL』を参照してください。ユーザーは、次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 (同時ユーザー) 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

デフォルト値は、次のとおりです。

```
<default
maxDbs           = " 8"
maxIdxPerDb      = " 25"
maxLocksPerIdx  = "100"

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

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



```
latchTimeout = "80"
/>
```

構文は、<default attribute=value.../> です。属性およびその意味は、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 (int>1)。

maxIdxPerDb

ロックできるデータベースあたりの索引数 (int>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 (int>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の試行の数、sWait はこれらの 2 つの試行の間の待ち時間 (int>1)。これらの 2 つのパラメーターは、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の試行の数、uWait はこれらの 2 つの試行の間の待ち時間 (int>1)。

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の試行の数、xWait はこれらの 2 つの試行の間の待ち時間 (int>1)。

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。したがって、ロックのための待ち時間の合計は、次のようにして計算できます。

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

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索引の場合 (実際にロックされた索引だけがリストされます)

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

ロック・スナップショットの表示

初めてテキスト索引がロックされる場合、メモリーはデータベースおよびロック・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、メモリーはロック・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、ロック・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。

関数 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除します。このコマンドの使用法の詳細に関しては、100 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 の時に開始されます。更新サービスは、テキスト索引を入手して、指定した時間に自動的に更新するために必要です。テキスト索引は、継続してユーザー表と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して、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します。

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

この例は、毎週の月曜日と水曜日の 12pm と 12am に更新サービスが起動し、索引 db2ext.title に実行すべき作業が存在するかをチェックすることを意味します。少なくとも 5 回の変更があって初めて、テキスト索引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、121 ページの『CREATE INDEX』を参照してください。

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
- ストアード・プロシージャで使用できるテキスト索引の作成
- 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』の内容を確認してください。

その他の索引作成の前提条件には、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、17 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

テキスト索引は、DB2 コントロール・センターを使用して作成し、保守することもできます。49 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

簡単にするために、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ぶことにします。

DB2 Net Search Extender コマンド行プロセッサを開始する

ヒント

索引の作成と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、

```
db2text ? CREATE INDEX
```

注

?, (,), *, ! および " は、システム・シェルによって解釈されます。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が必要です。

たとえば、UNIX では次のようにします。

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

データベースの使用可能化

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 データベース 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、153 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、複数の表およびさまざまなビューも作成されます。それらは、次のものです。

- `db2ext.dbdefaults` は、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` は、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` は、索引構成パラメーターを示します。
- `db2ext.textindexes` は、すべてのテキスト索引を追跡するカタログ・ビューです。
- `db2ext.proxyinformation` は、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、187 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。

データベースは、使用可能になると、ユーザーが使用不可にするまで、使用可能の状態が続きます。

データベースのディセーブリング

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 データベース
権限	データベースに対する DBADM

DB2 Net Search Extender が使用できるようにデータベースを準備するときに、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、準備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。

接続データベースを使用不可にするには、次のコマンドを入力します。

```
db2text DISABLE DATABASE FOR TEXT
```

データベースを使用不可にすると、次のオブジェクトが削除されます。

データベースの使用可能化

- 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、次のコマンドを入力します。

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

注

force を使用しないでデータベースを使用不可にした場合、既存の索引は除去されません。データベースを使用不可にする前に索引をドロップするか、force オプションを使用して DISABLE を呼び出してください。

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 ... (例を参照)
権限	表に関する 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
- データ・リンク・データ・タイプ

また、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に関しても要件は異なります。これらの要件については、以下のセクションで説明します。

テキスト索引を作成すると、以下のオブジェクトが作成されます。

- ユーザー表内の変更されたすべての行の追跡に使用される内部表。 **Recreate index on Update** オプションを選択すると、ログ・テーブルは作成されないので注意してください。
- テキスト索引の更新時の問題に関する情報を収集するために使用されるイベント表。

- 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、次回の索引作成時の索引同期化のために必要です。

ログ・テーブルを作成し、テキスト索引を、ビューまたはニックネーム表でなく、基本表に作成した場合のみ、トリガーが作成され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・スペースの利用を最適化するために、CREATE INDEX コマンドを使用して、表に使用する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際、トリガーが開始されず、ロードされた文書のインクリメンタル索引作成ができないので注意が必要です。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活動化します。

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。

索引作成のためにキューに入れられた文書が見つからない場合のように、索引作成時にエラーが発生した場合には、**索引更新イベント**がイベント表に追加されます。この説明は 193 ページの『イベント・ビュー』をご覧ください。

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。この説明は 46 ページの『テキスト索引をドロップする』をご覧ください。

テキスト索引とデータベースを同期させるには、次のコマンドを呼び出します。

```
db2text update index
```

同期処理が終了するまで、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

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、DB2 はデータを変換しないので、文書は元のコードページ (CCSID) を保持します。2 つの異なるコード・ページが存在する場合、使用されるコード・ページが、データベースのコード・ページまたは CREATE INDEX コマンドで指定したコード・ページのいずれかになるので、テキスト索引の作成時に問題が生じる可能性があります。

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 indexname = 'COMMENT'
        and indexschema = 'DB2EXT'
```

また、1 つのテキスト索引内に、異なるコード・ページの文書をもつこともサポートされていないので注意してください。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。ix ページの『関連情報』を参照してください。

他の文字データ・タイプの索引作成に関しては、このような問題は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

DB2 Net Search Extender を使用して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フォーマット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、121 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. 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 CLOB(10M) 値などとして戻す、uncompress などという名前のユーザー定義関数 (UDF) を作成します。

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、'socks' 上のポート 'proxy'、およびタイムアウトの秒数値を含めます。たとえば、

```
insert ('hostname', '80' 'proxy', 10)
```

テキスト索引の作成

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す。

- 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 V8 の資料を参照してください。基本的には、データベース・マネージャー構成を調整する必要があります。

DATALINK UDF は Blob (100 KB) を戻します。戻されたサイズは、DB2EXTDL を使用して変更できます。また、それに従って、データベース・マネージャー構成パラメーター `java_heap_sz` を更新する必要があります。

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

ストアード・プロシージャ検索を使用するには、ハイパフォーマンスを達成し、すべての指定したデータをメイン・メモリーに移動するために、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。

ただし、ストアード・プロシージャを使用した検索のための最初の索引更新前に表に文書をロードしておき、文書のない表の索引を更新しないようにする必要があります。これにより、より良い索引作成のパフォーマンスおよびキャッシュ・メモリー要件のしっかりした予測を立てる機会が得られます。

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。このデータは、CREATE INDEX コマンドで、キャッシュ表オプションを使用して定義する必要があります。次に、ACTIVATE 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

ビューに関してテキスト索引が作成されていない場合は、そのテキスト索引は、SQL スカラー検索関数でも使用できます。

ストアード・プロシージャで使用するテキスト索引を作成する場合は、いくつかのパラメーターを決定し、計算しておく必要があります。それらのパラメーターは以下のとおりです。

- キャッシュのタイプ
- どのように索引を更新するか
- Net Search Extender で使用できる最大メモリー量。MAXIMUM CACHE SIZE
- インクリメンタル更新の場合のみ。PCTFREE、つまり、以後の文書更新に必要な空きメモリーの量

2 種類の使用可能なキャッシュがあります。

- 一時キャッシュは、DB2TEXT ACTIVATE CACHE コマンドにより再作成する必要があり、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。これは、永続的キャ

ストアド・プロシージャ検索で使用できるキャッシュに入れられたテキスト索引の作成

ッシュの活動化より長い時間を取ります。特に大きな索引の場合にはそうです。しかし検索においては、少し良いパフォーマンスが得られています。

- **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 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 (下記参照) 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活動化する必要があります。そうでないと、次の DB2TEXT ACTIVATE CACHE コマンドがキャッシュをスクラッチから再作成してしまいます。

テキスト索引を更新する 2 つの方式があり、それぞれ、異なる係数を持ちます。

- 更新のオプションに索引の再作成を指定しない方法 (インクリメンタル更新オプションとも呼ばれます)。
 - 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、活動化された索引では、キー列を変更しないようにする必要があります。
- 更新オプションで索引の再作成を指定する方式 (更新のたびに索引を再作成する際の指定です)。ul>- 可能な場合は、キャッシュ列の式で VARIABLE 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。
- 最初の索引の活動化以後に、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、MAXIMUM CACHE SIZE および PCTFREE を決定するのに役立つ 2 つの SQL 関数があります。

- インクリメンタル更新および再作成更新の場合は以下のようにします。

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

推奨される MAXIMUM CACHE SIZE 値が戻ります。ユーザー表への今後の文書の追加が大幅に平均に影響するというのでない限り、平均の行長さパラメーターは、ユーザー表の SQL 照会

```
SELECT AVG(LENGTH(cache column1) + ... + LENGTH(cache column n))
```

により計算できます。キャッシュ列の数は、DB2TEXT CREATE 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。

- インクリメンタル更新のみの場合は以下のようにします。

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

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。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。

ストアード・プロシージャ検索で使用できるキャッシュに入れられたテキスト索引の作成

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。

索引が各更新ごとに再作成される場合は、PCTFREE 値は 0 に設定する必要があります。

適切なパラメーターを決定後、次のようにして、索引およびキャッシュ表を作成します。

```
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ます。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

この索引での検索を可能にするには、索引を更新してから、活動化し、指定した表キャッシュ式をデータベースからメモリーにコピーする必要があります。

ACTIVATE または UPDATE の操作中に MAXIMUM CACHE SIZE または MAXIMUM NUMBER OF DOCUMENTS 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。

- インクリメンタル更新の MAXIMUM CACHE SIZE。次の順序で DB2EXT 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。
- MAXIMUM NUMBER OF DOCUMENTS。インクリメンタル更新の場合は、DEACTIVATE CACHE および ACTIVATE CACHE RECREATE のコマンドを使用する。あるいは、索引を再作成する。

文書の頻繁な更新が予想され、CACHE TABLE 文節のキャッシュ列式に固定サイズ・データ・タイプの使用を考慮してください。たとえば、

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

これにより、更新操作の間、同じキャッシュ・ストレージの使用が可能になります。

テキスト索引のキャッシュの活動化および非活動化

キャッシュを活動化する前に、不正確な PCTFREE 計算を防ぐため、インクリメンタル更新を保留する必要があります。

テキスト索引を活動化するには、以下のようになります。

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

これによって、指定したキャッシュの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

索引が活動化されているときに `update index` を呼び出すと、キャッシュの表も更新されます。削除した文書はキャッシュのスロットを消費するため、`PCTFREE` を十分大きな値に設定してください。

`ACTIVATE INDEX` 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。永続的キャッシュが使用されていた場合、新規の活動化は素早いものです。

注

永続キャッシュを活動化していないときに更新を行うと、まず、永続キャッシュがドロップされ、次に活動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活動化する必要もあります。

どれだけのメモリーが残っているかをチェックするには、以下のコマンドを実行してください。

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

このコマンドは、索引が活動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

ストアード・プロシージャを使用する場合、ビューに基づくテキスト索引を作成できます。ただし、ビュー上でトリガーを作成できないため、1つの主要な欠点として、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、インクリメンタル索引更新の場合、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説明します。

1. 次のように基本表を作成します。

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

2. 項目をいくつか追加します。

ビューに基づくテキスト索引

```
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 ')"
```

次に、ビューを作成します。

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

3. それから、索引を作成し、更新し、活動化します。

```
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 myindexview for text"
```

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ることに注意してください。さらに、正しいログ表を作成するには、ビューに基づく索引のキー欄を指定する必要があることにも注意してください。

4. 以下のコマンドを使用して、表を更新します。

```
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,'Guenter','shares the office with Raiko')"
```

5. 次にログ表を更新します。ログ表の名前を取得するには、次のようにします。

```
db2 select "INDSCHEMA,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

ここで、ログ表に項目を追加します。

```
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。

6. 以下のようにして、索引をもう一度更新します。

```
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します。
- 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・ディスクを使用します。
- 小さな主キーを使用します。
- 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ーがあることを確認してください。メモリーが不十分な場合、オペレーティング・システムは代わりにページング・スペースを使用するので、検索パフォーマンスが低下します。

異なるプラットフォームのメモリー所要量の構成については、185 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。

- 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、初期更新（このパラメーターは使用されない）を除いて、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。

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入力する方法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。

- テキスト索引の更新と再編成
- テキスト索引の変更
- 索引更新イベントを削除する
- テキスト索引をドロップする
- 索引状況を表示する

上記のタスクをいつでも、どのようなシーケンスでも実行できます。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、場合に、トレース機能をオンにする方法についても説明します。

テキスト索引の情報を表示するコマンドについては、187 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。

テキスト索引は、DB2 コントロール・センターを使用して、保守することもできます。68 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および更新した後に、テキスト文書をデータベースに追加する場合、またはデータベース内の既存の文書を変更する場合、索引の内容とデータベー

テキスト索引の更新と再編成

スの内容を同期化するために、文書に索引を付ける必要があります。テキスト文書をデータベースから削除する場合、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 (INCREMENTAL UPDATE オプションとも呼ばれる) が指定されずにテキスト索引が作成されていれば、どの文書が新規であり、変更され、削除されたかに関する情報は、トリガーによって自動的に内部ログ表に保管されます。

ログ表で参照された文書は、次回に索引更新が行われた時に索引付けされます。ビューに基づくテキスト索引についての詳細は、41 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。この場合、ログ表またはトリガーは作成されません。

索引は、一般に時々更新されます。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、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、文書のサイズ、およびプロセッサの能力などの、多くの要素によって異なります。

注

ロールバックまたはデッドロックの状態が発生する可能性があるため、DB2 表に関する更新および変更トランザクションの頻度が高くなる時には注意してください。DB2 表に関して長いトランザクションを実行する場合にも、この状態が発生する可能性があります。

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	表に関する CONTROL

次の例は、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。

指定された `MANUAL REORGANIZATION` で列を頻繁に更新する場合、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
```

手動の再編成が必要かどうかを判別するには、次のようにして、`db2ext.textindexes` ビューを照会してください。

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

ただし、`CREATE INDEX` 時に `AUTOMATIC REORGANIZE` を指定した場合には、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	更新の頻度、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	<code>ALTER INDEX</code>
権限	表に関する <code>CONTROL</code>

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、あるいはディレクトリー、`MAXIMUM CACHE SIZE` または `PCTFREE` などのキャッシュ特性を変更します。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。索引の更新または検索が実行中の場合には、索引が現在ロックされていて、変更できないことを示すエラー・メッセージが表示されます。

次の例は、索引の更新頻度を変更して、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、索引を更新するようにします。

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

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
```

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	表に関する 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。

次の例は、共通索引表の索引からメッセージを削除するものです。

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

テキスト索引をドロップする

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	表に関する CONTROL

例：

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

索引をドロップした場合、次のオブジェクトもドロップされます。

- テキスト索引
- 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。

テキスト索引状況を表示する

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、作成した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るには、以下を使用します。

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

現在使用できる索引、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、次のようにします。

```
db2 select indschema, indname, tabschema, tabname,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 187 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。

索引のバックアップとリストア

使用可能データベースおよび DB2 Net Search Extender が作成したテキスト索引をバックアップし、リストアできます。

バックアップするには、次のようにします。

1. DB2 Net Search Extender が作成した索引と保管先を検出します。これを実行するには、DB2 Net Search Extender db2ext.textindexes ビューに関して SELECT ステートメントを実行します。

```
db2 select indexschema, indexname, 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そのサブディレクトリーをバックアップします。

4. DB2 Net Search Extender サービスを再始動します。

```
db2text start
```

テキスト索引をドロップする

リストアするには、次のようにします。

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
```

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、いろいろな 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)
- テキスト索引オブジェクト (Text 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、コマンド行を使用します。説明については、31 ページの『第 6 章 テキスト索引の作成と保守』および 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。

注

この例および 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だけを説明します。DB2 コントロール・センターの使用法については、ix 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** を選択すると、追加のコマンドがポップアップ・メニューに表示されます。追加のコマンドは次のとおりです。

- DB2 Net Search Extender インスタンス・サービスの開始。このコマンドによって、インスタンス・サービスがまだ開始されていない場合は、開始されます。
- DB2 Net Search Extender インスタンス・サービスの停止。



図7. 「Net Search Extender サービス」ダイアログの停止

確認ダイアログ・ボックスが表示されます。「強制 (FORCE)」オプションを使用可能にするには、チェック・ボックスを使用してください。「OK」ボタンをクリックします。

- インスタンス状況 (Instance status)。ダイアログによってインスタンスの状況が表示されます。

99 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。

データベースの使用可能化

インスタンス・オブジェクトをクリックすると、インスタンスに使用可能なデータベースがオブジェクト・ツリーに表示されます。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** を強調表示すると、追加のコマンドが拡張メニューに表示されます。追加のコマンドは次のとおりです。

- テキストのデータベースの使用可能化 (Enable the database for text)。データベースが使用可能になっていない場合は、ダイアログがポップアップ表示されます。「**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す。
- テキストのデータベースの使用不能化 (Disable the database for text)。データベースが使用不可になっていない場合は、ダイアログがポップアップ表示されます。「**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す。

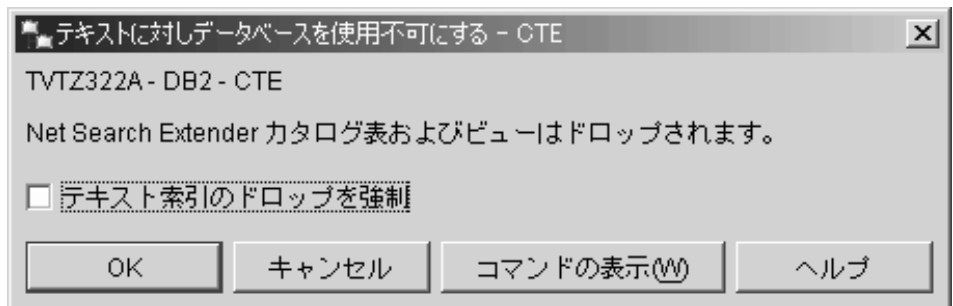


図 8. 「テキストのデータベースの使用不可化 (Disable database for text)」ダイアログ

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。

- DATALINK の戻りサイズの変更 (Change the datalink return size)。ダイアログがポップアップ表示されます。フィールドに現在の戻りが表示されます。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。

105 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 (Show Command)**」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

データベース・オブジェクトの下に、テキスト索引オブジェクトが表示されます。テキスト索引オブジェクトをクリックして、テキスト索引を目次ペインに表示します。

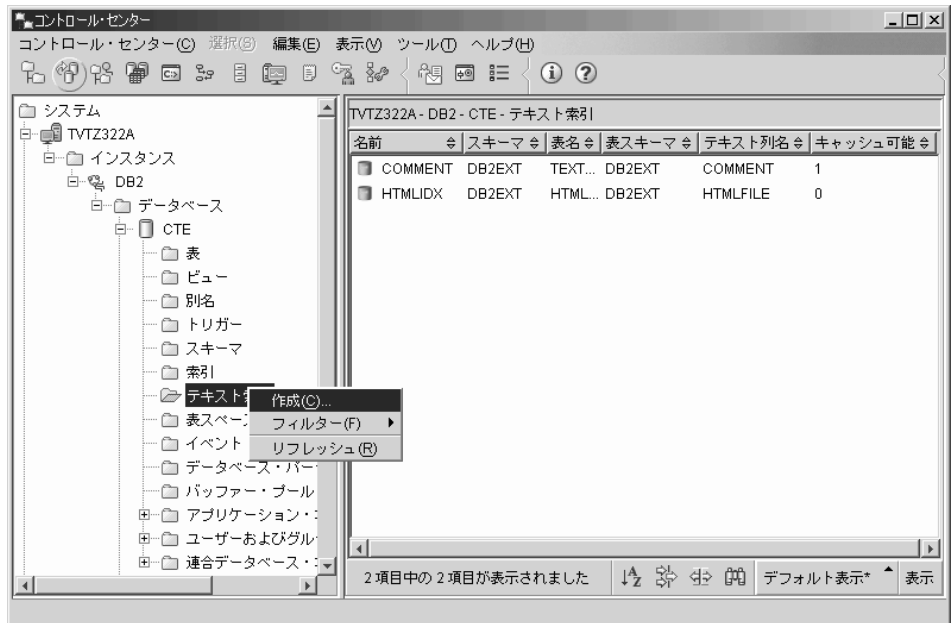


図9. DB2 コントロール・センター 2

テキスト索引オブジェクトを右クリックすると、次のコマンドを含むポップアップ・メニューが表示されます。

- 作成 (Create)。テキスト索引を作成するウィザードが表示されます。詳しくは、次のセクション 53 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 (Filter)。ダイアログを使用すると、コントロール・ペイン・ビューに表示するテキスト索引オブジェクトを決定できます。
- リフレッシュ (Refresh)。オブジェクト・ツリーおよびコントロール・ペインに表示されている情報が最新表示されます。

以前に作成したテキスト索引を保守する方法の詳細については、68 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 (right click)」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、「**選択済み (Selected)**」メニュー・コマンドをクリックし、「**Net Search Extender**」を強調表示して、関係のある拡張コマンドにアクセスします。

索引作成の前に、23 ページの『第 4 章 計画』で説明した最初のステップ、関連する文書フォーマットと文書モデル、CCSID、およびディスク・スペース要件について考慮されていることを確認してください。

その他の索引作成に関する必要条件には、以下のものがあります。

- DB2 Net Search Extender Instance 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

「索引の作成 (Create Index)」コマンドを選択することによって、「テキスト索引の作成ウィザード (Create Text Index Wizard)」がポップアップ表示されます。このウィザードを多くのパネルで使用して、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、「**終了 (Finish)**」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 (Next)**」ボタンをクリックします。

「名前 (NAME)」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を入れる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。索引の管理表は、管理表スペース上に作成されます。

テキスト索引の作成

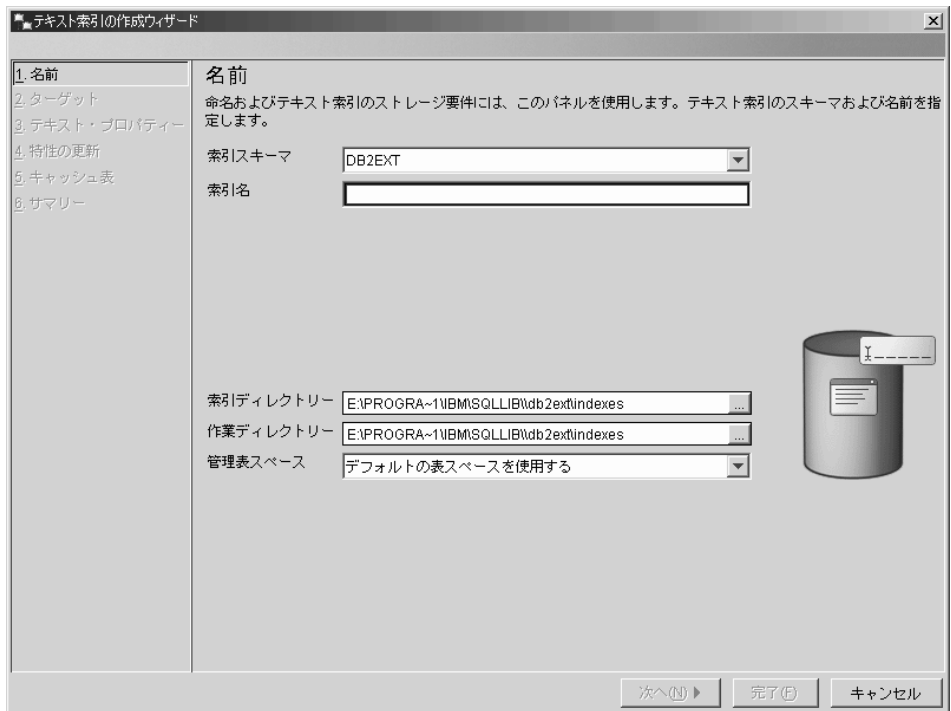


図 10. テキスト索引の作成ウィザード (Create Text Index Wizard): 「名前 (Name)」 パネル

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 (Name)」 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 (Index schema)	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、索引特有の管理表の DB2 スキーマ名として使用されます。
索引名 (Index name)	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的に識別します。

表1. 「名前 (Name)」 パネルのテキスト・フィールド (続き)

索引ディレクトリー (Index directory)	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ー・パスを指定します。このディレクトリーは、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している必要があります。
作業ディレクトリー (Work directory)	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するために使用する作業ディレクトリーを指定します。このディレクトリーは、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している必要があります。
管理表スペース (Administration tablespace)	オプション	デフォルトの表スペースを使用	索引用に作成する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・グループで定義する必要があります。

「ターゲット (Target)」 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関数を使用すれば、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、数値属性を指定します。

テキスト索引の作成

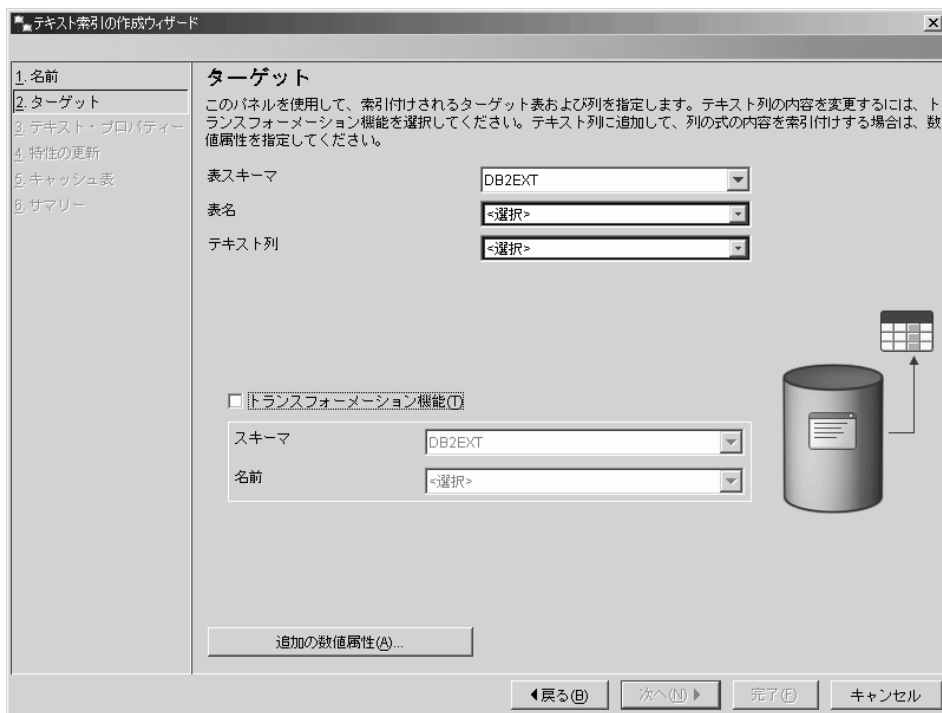


図 11. テキスト索引の作成ウィザード (Create Text Index Wizard): 「ターゲット (Target)」パネル
次は、パネル内の各フィールドについての説明です。

表 2. 「ターゲット (Target)」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 (Table schema) (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 (Table name) (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、主キーがある必要があります。

表 2. 「ターゲット (Target)」 パネルのテキスト・フィールド (続き)

テキスト列 (Text column) (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。
変換関数 (Transformation function)	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。
トランスフォーメーション関数: スキーマ (Transformation function: Schema)	必須 (関数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関数: 名前 (Transformation function: Name)	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 (table schema) (1)」、 「表名 (table name) (2)」、 および 「テキスト列 (text column) (3)」 のフィールドは、この順序でのみ指定できることに注意してください。

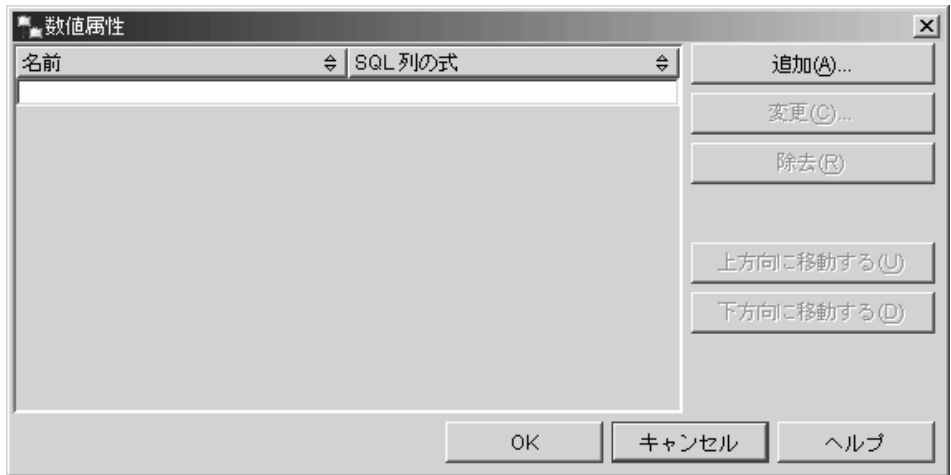


図 12. 「数値属性 (Numeric Attributes)」 ダイアログ

属性を表示または追加するには、「数値属性 (Numeric Attributes)」 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 (Add)」 ボタンをクリックします。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。

数値属性の説明

テキスト列に加えて、列式に索引付けしたい場合には、数値属性を指定します。たとえば、テキスト列に加えて、タイプ `TIMESTAMP` の列日付に索引付けしたい場合は、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。たとえば、`WHERE numattrib = 123 AND contains('...')` のようにします。ストアード・プロシージャを使用して検索する場合、テキスト文書への追加情報を検索したいときは、数値属性を指定する必要があります。

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。

「テキスト・プロパティ (Text Properties)」 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

い。文書が GPP、HTML 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。

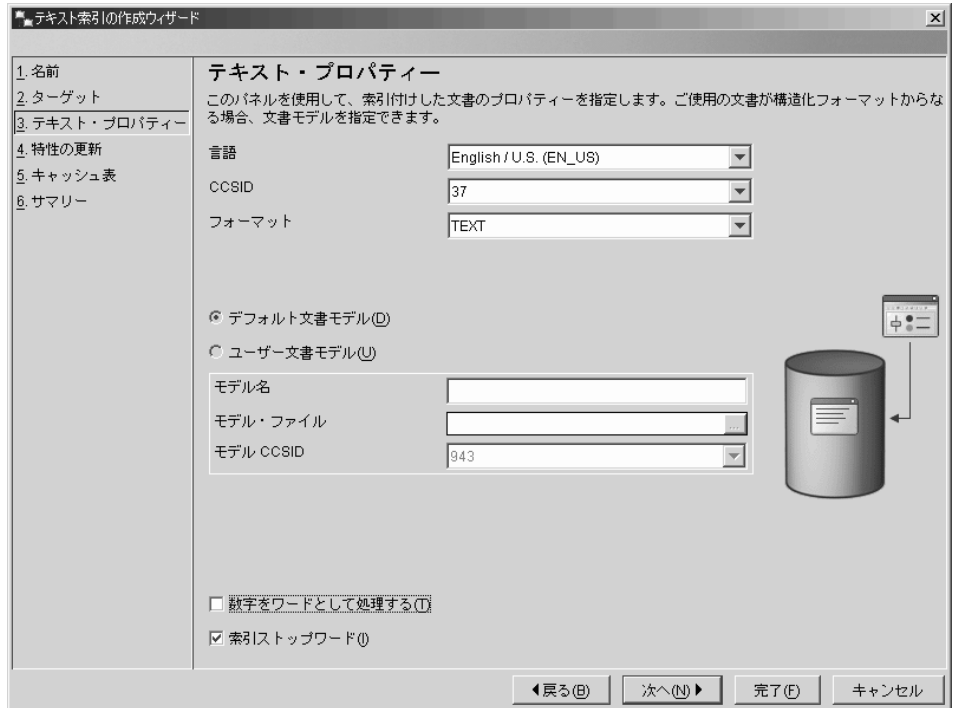


図 13. テキスト索引の作成ウィザード (Create Text Index Wizard): 「テキスト・プロパティ (Text Properties)」パネル

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 (Text Properties)」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 (Language)	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 (Format)	オプション	TEXT	HTML、XML、TEXT および 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 (Text Properties)」 パネルのテキスト・フィールド (続き)

デフォルト文書モデル (Default Document Model)	オプション	使用可能	デフォルト文書モデルを使用します。
ユーザー文書モデル (User Document Model)	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 (Model name)	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 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 (Model file)	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 (Model 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。
数値をワードとして処理 (Treat numbers as words)	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 (Index stopwords)	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>sqllib/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。

「更新特性 (Update characteristics)」 パネル

このパネルによって、索引をインクリメンタル更新するか、またはゼロ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。

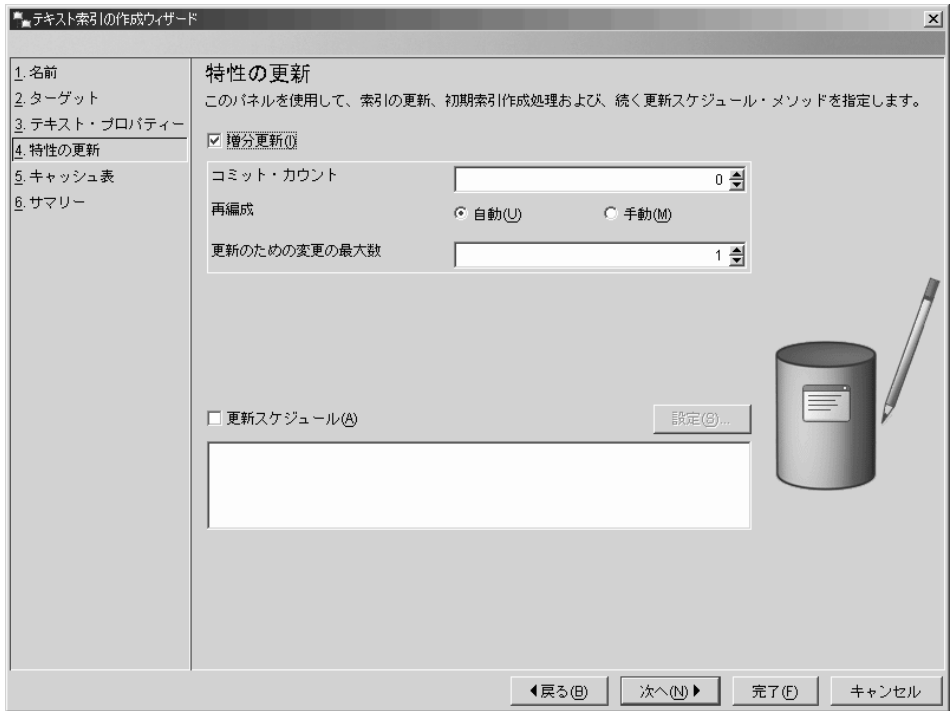


図 14. テキスト索引の作成ウィザード (Create Text Index Wizard): 「更新特性 (Update Characteristics)」 パネル

次は、パネル内の各フィールドについての説明です。

表 4. 「更新特性 (Update Characteristics)」 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 (Incremental update)	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。選択しないと、更新操作の実行時に索引が再作成されます。
コミット・カウント (Commitcount)	オプションまたは必須	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。

テキスト索引の作成

表4. 「更新特性 (Update Characteristics)」 パネルのテキスト・フィールド (続き)

再編成自動または手動ラジオ・ボタン (Reorg automatic or manual radio button)	オプションまたは必須	使用可能 / 使用不可	索引再編成が自動または手動で完了します。
更新の変更の最小数 (Minimum number of changes for Update)	オプション	1	指定した時刻に索引をインクリメンタル更新するための、テキスト文書に対する変更の最小数を指定します。
更新スケジュール (Update schedule)	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、「設定値 (Settings)」ボタンをクリックします。このボタンは、「自動更新 (Automatic Update)」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。ダイアログが表示されます。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ることに注意してください。

「キャッシュ表 (Cache table)」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でき、ストアード・プロシージャを使用してキャッシュを検索できます。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。

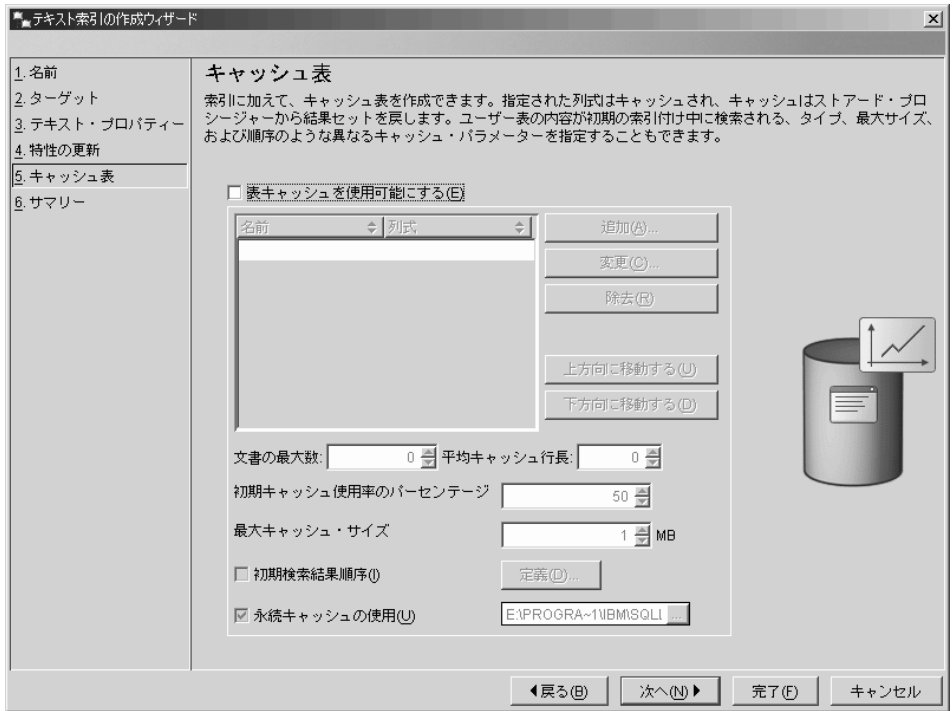


図 15. テキスト索引の作成ウィザード (Create Text Index Wizard): 「キャッシュ表 (Cache Table)」パネル

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
表キャッシュの使用可能化 (Enable table cache)	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 (表キャッシュの使用可能化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 (Maximum number of documents)	必須	表の行数	以下の例を参照してください。

テキスト索引の作成

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)

平均キャッシュ行長 (Average cache row length)	必須	N/A	以下の例を参照してください。
初期キャッシュ使用率のパーセント (Percentage of initial cache utilization)	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 (Maximum cache size)	オプション	N/A	索引の活動化時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、活動化は失敗します。
初期検索の結果順序 (Initial search result order)	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。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ことによって、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。
永続キャッシュの使用 (Use persistent cache)	オプション	使用可能	このオプションは、非活動化またはシステム・リポート後の高速活動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 (Percentage of initial cache utilization)」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 (Maximum cache size)」は、キャッシュの活動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、表にある実際の文書数、予想更新数、およびキャッシュに入れたい SQL 式の平均サイズによって異なります。

「初期キャッシュ使用率のパーセンテージ (Percentage of initial cache utilization)」および「最大キャッシュ・サイズ (Maximum cache size)」は直接入力できます。あるいは、「文書の最大数 (Maximum number of documents)」または「平均キャッシュ行長 (Average cache row length)」のフィールドに値を入力するたびに再計算されます。

「文書の最大数 (Maximum number of documents)」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 (Average cache row length)」が計算されます。平均したこの値が小さいとわかった場合は、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 10000 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、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 (Define)**」ボタンをクリックします。このボタンは、「初期検索の結果順序 (Initial Search Result Order)」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。すべての指定した SQL 列式を示すダイアログが表示されます。結果順序を追加するには、「**追加 (Add)**」ボタンをクリックします。ダイアログで、SQL の結果順序を指定します。

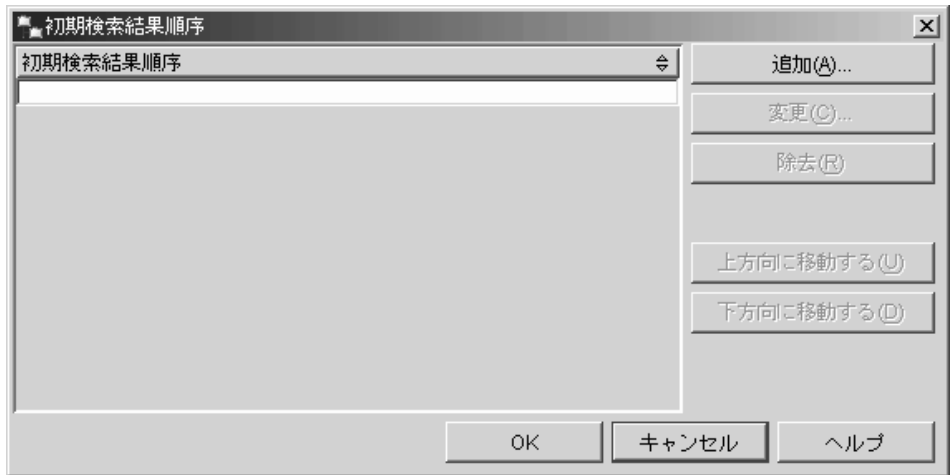


図 16. 「初期検索結果順序 (Initial Search Result Order)」ダイアログ

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。

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

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



図 17. 「列式の変更 (Change column expression)」ダイアログ

「サマリー (Summary)」パネル

このパネルは、以前に選択したパラメーターについて概説します。

テキスト索引の作成



図 18. テキスト索引の作成ウィザード (Create Text Index Wizard): 「サマリー (Summary)」 パネル

「コマンドの表示 (Show Command)」 ボタンをクリックして、「終了 (Finish)」 ボタンをクリックした場合に実行されるコマンドを表示します。これによって、テキスト索引が作成されます。

テキスト索引の保守

作成したテキスト索引を保守するには、コントロール・ペインでテキスト索引を選択し、「選択 (Select)」メニュー・コマンドをクリックします。以下のコマンドを使用できます。

- テキスト索引の変更 (Alter a text index)
- テキスト索引のドロップ (Drop a text index)
- テキスト索引の更新 (Update a text index)
- 索引イベントの表示 (Show index events)
- 索引キャッシュの活動化 (Activate index cache)
- 索引キャッシュの非活動化 (Deactivate index cache)
- 索引状況の表示 (Show index status)

ここで、それぞれのコマンドを説明します。索引イベントの活動化および索引イベントの非活動化は、キャッシュ・オプションを使用して索引を作成する場合にだけ表示されることに注意してください。

注

各パネルの情報は、53 ページの『テキスト索引の作成』にあるサンプルに対応しています。

ダイアログの「**コマンドの表示 (Show Command)**」ボタンを使用すると、代わりにコマンド行が表示されます。

「索引の変更 (Alter index)」ダイアログ

「**変更 (Alter)**」コマンドを選択すると、ダイアログが表示されます。このダイアログには、以前に選択したテキスト索引のパラメーターについて概説する、一連のタブ・パネルが含まれています。このパネルで、必ずしもすべての構成を変えることはできません。

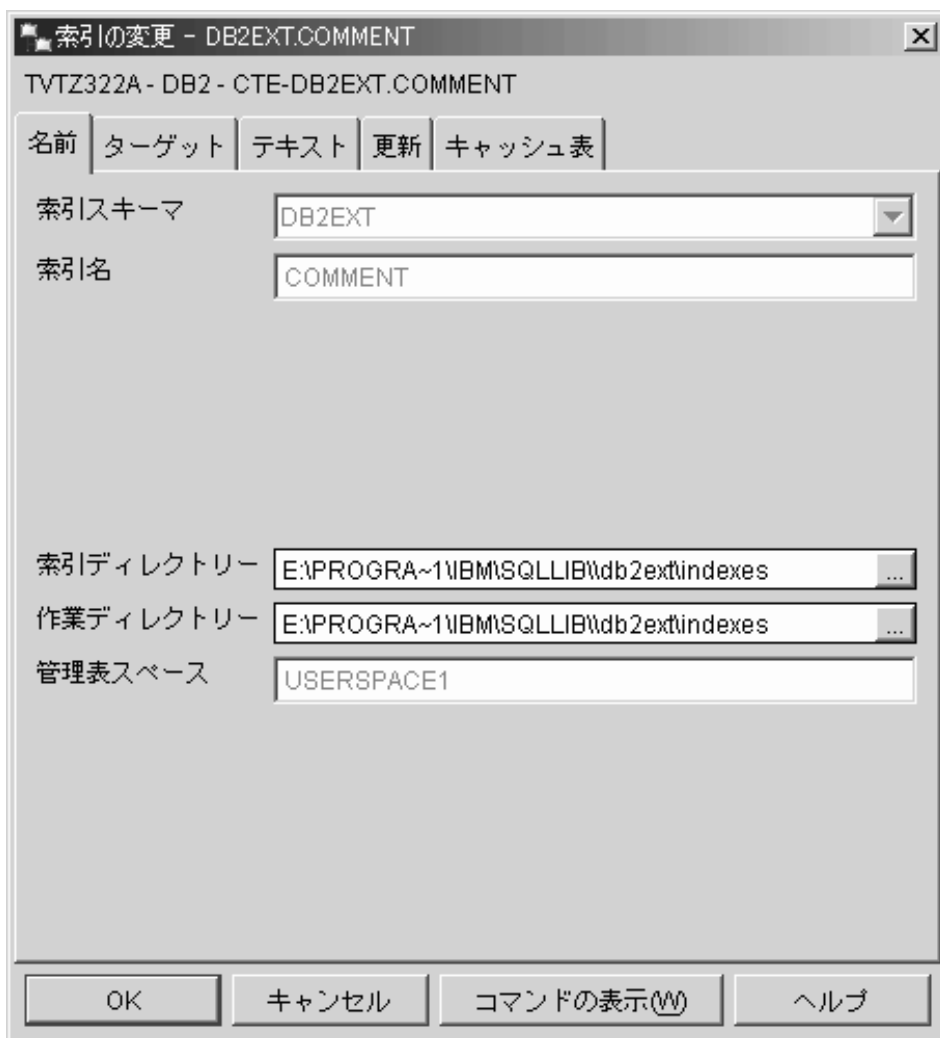


図 19. 「索引の変更 (Alter Index)」 ダイアログ: 「名前 (Name)」 タブ

「名前 (Name)」 ダイアログは、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 (Target)」 ダイアログは、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 (Text)」 ダイアログは、テキスト文書の構成を表示します。これらの設定値は変更できません。

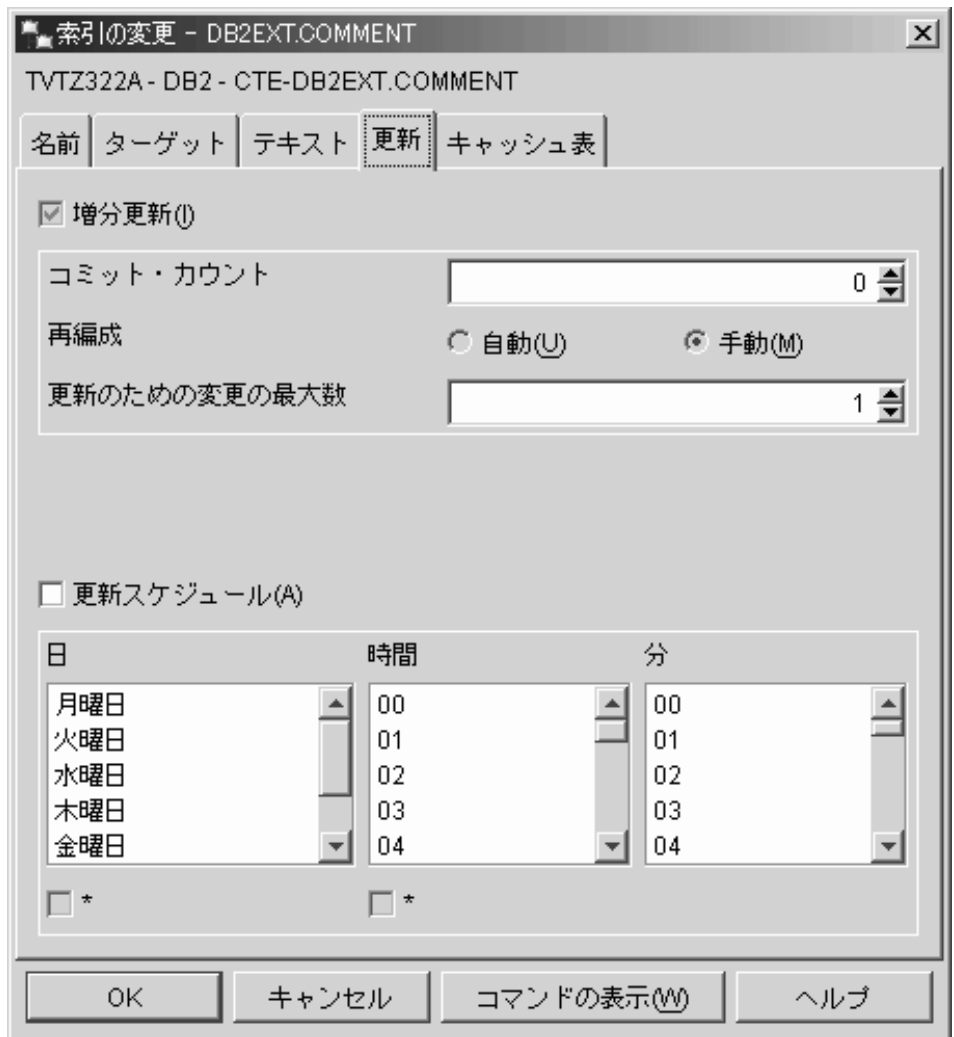


図 20. 「索引の変更 (Alter Index)」ダイアログ: 「更新 (Update)」タブ

「更新 (Update)」ダイアログは、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、コミット・カウントおよび変更の最小数も変更できます。



図 21. 「索引の変更 (Alter Index)」ダイアログ: 「キャッシュ表 (Cache Table)」タブ

「キャッシュ表 (Cache Table)」ダイアログは、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を変更することもできます。

「索引のドロップ (Drop index)」ダイアログ

「ドロップ (Drop)」コマンドを選択すると、確認ダイアログが表示されます。索引を選択し、「OK」ボタンをクリックします。



図 22. 「索引のドロップ (Drop Index)」 ダイアログ

「索引の更新 (Update index)」 ダイアログ

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



図 23. 「索引の更新 (Update Index)」 ダイアログ

この「更新 (update)」ダイアログで、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、「更新最小数の使用 (Using update minimum)」チェック・ボックスを選択してください。索引を再編成する必要がある場合には、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 (Index events)」ダイアログ

「索引イベントの表示 (Show index events)」コマンドを選択すると、ダイアログが表示されます。

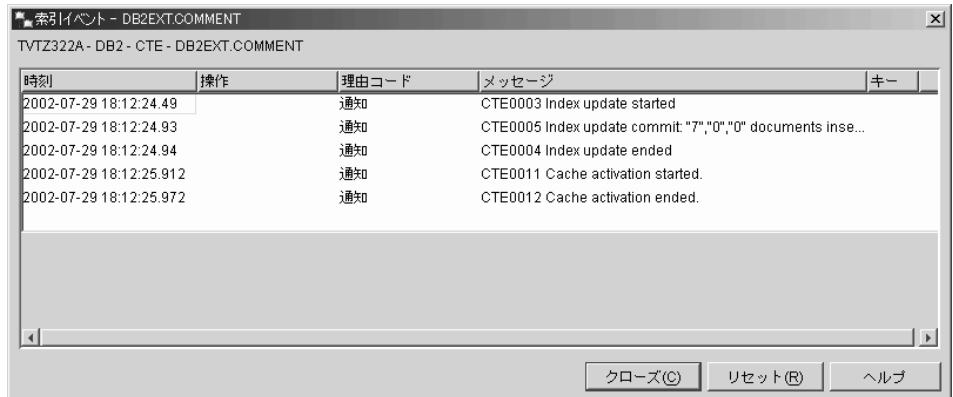


図 24. 「索引イベント (Index Events)」 ダイアログ

イベント表の内容が表示されます。索引イベントをクリアするには、「リセット (Reset)」 ボタンをクリックしてください。

「キャッシュの活動化 (Activate the cache)」 ダイアログ

「索引メモリの活動化 (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。

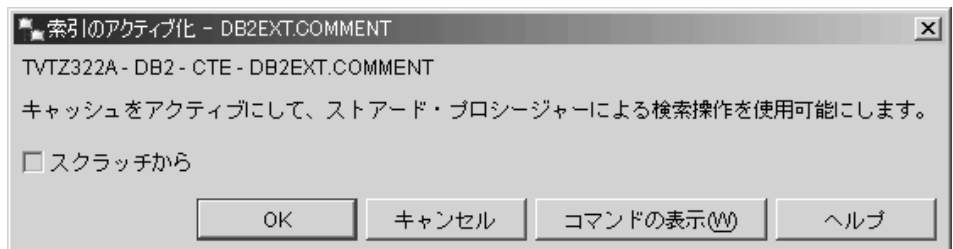


図 25. 「キャッシュの活動化 (Activate the cache)」 ダイアログ

「活動化 (activate)」 ダイアログで、新たなキャッシュの作成を指定できます。

「キャッシュ表の非活動化 (Deactivate the cached table)」 ダイアログ

「索引メモリの非活動化 (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。

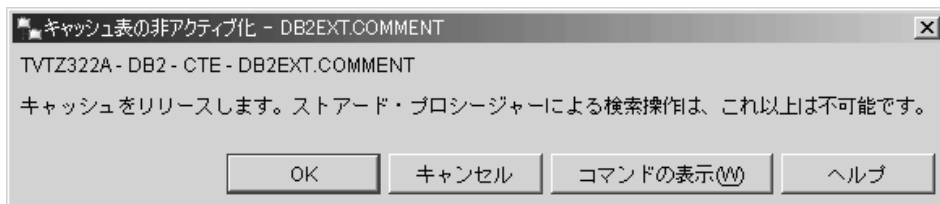


図 26. 「キャッシュ表の非活動化 (Deactivate the cached table)」ダイアログ

このダイアログは、非活動化アクションを確認します。

「索引状況 (Index Status)」ダイアログ

「状況の表示 (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 (Index Status)」ダイアログ

このダイアログは選択した索引の状況を表示します。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方式が 3 つあります。それらは、以下の通りです。

- 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。これらの関数は、SQL で通常使用可能な関数に加えて提供されます。
- 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、この章では以下のことを説明します。

- 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 153 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 145 ページの『第 14 章 検索引き数の構文』を参照してください。

検索関数および検索引き数では、サンプル表 TEXTTAB から、COMMENT と呼ばれるサンプル・テキスト索引が作成されています。

このセクションの最後に、追加の検索構文例も示してあります。

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、145 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、この章では以下のことを説明します。

- SQL 表値関数を使用するテキストの検索。
- 検索引き数の指定については、157 ページの『DB2EXT.TEXTSEARCH』のパラメーターの説明を参照してください。

その後、検索のパフォーマンスに関する問題を簡単に述べます。

検索

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

?, (), *, ! および " は、システム・シェルによって解釈されます。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が必要です。

たとえば、UNIX では次のようにします。

```
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、DB2 Net Search Extender と共に提供される SQL スカラー検索関数を使用して、Net Search Extender によって索引付けされたテキストを含む DB2 データベースを検索する方法について説明します。このセクションでは、次のことを行う方法について説明します。

- 関数 CONTAINS を使用して、照会する
- 関数 NUMBEROFMATCHES を使用して、テキスト文書内で一致がいくつ見つかったかを判断する
- 関数 SCORE を使用して、見つかったテキスト文書の適合性を入手する

これらの関数の構文の説明は 153 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 enable を検索します。

ヒント

表または列に関して、大文字小文字混合の ID を作成した場合、これらの名前は二重引用符で囲む必要があります。たとえば、

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

UNIX コマンド行から DB2 select ステートメントを指定した場合には、オペレーティング・システムのコマンド行パーサーによって、二重引用符などの特殊文字がコマンド・ストリングから除去されるので、バックスラッシュ (\) を使用して、これらの特殊記号をマスクする必要があります。たとえば、

```
WHERE CONTAINS(COMMENT, "\"book\"") = 1
```

検索と見つかった一致数の戻し

NUMBEROFMATCHES 関数を使用して、各テキスト文書で検索基準が見つかった頻度を判断します。

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

NUMBEROFMATCHES によって整数値が戻されます。

検索と検出したテキスト文書のスコアの戻し

SCORE は、この文書が、検出された他の文書と比較して、検索基準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。SCORE 関数を使用して、検出された文書のスコアを入手できます。

例を次に示します。

```
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

ビューに基づいて作成した索引は、CONTAINS、SCORE および NUMBEROFMATCHES 検索関数では使用できないことに注意してください。

SQL 検索引き数の指定

CONTAINS、NUMBEROFMATCHES、および SCORE で使用する検索引き数。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。

これらの関数の構文の説明は 145 ページの『第 14 章 検索引き数の構文』を参照してください。

任意のシーケンスでの用語検索

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。論理項で、検索語は OR 演算子によって接続されます。

ブール演算子 AND と OR を使用する検索

(83 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。)

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。たとえば、

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

複数の検索語を、ブール演算子を使用して結合できます。

```
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した場合、DB2 Net Search Extender は左から右へ評価しますが、論理 AND 演算子 (&) は、論理 OR 演算子 (|) より強く結合します。たとえば、括弧を含まない場合、

```
"book" & "pullitzer" | year" & "author"
```

は、以下のように評価されます。

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

したがって、次の例では、括弧を使用する必要があります。

```
"book" & ("pullitzer" | 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を、ブール演算子で結合した場合、

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

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" | "pullitzer") & "book"
```

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。この方法は、検索によって検出されるテキスト文書の数を増加します。

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。検索語の前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
```

この検索語が、たとえば、“username”、“file_name”、および“name”を含むテキスト文書を検出します。

- _ は、検索語内の 1 文字を表します。次の例は、“CLOB” および “BLOB” 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、ESCAPE キーワードを使用してエスケープ文字を識別してください。

たとえば、

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

この例のエスケープ文字は "!" です。

固定シーケンスでの用語検索

次の例のように “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発生している場合にのみ、これらの用語を検出します。

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

任意のシーケンスでの用語検索

任意のシーケンスで用語を検索するには、各用語をコンマを使用して区切ります。

```
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発生してい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
```

一緒に発生している 3 つ以上のワードも検索できます。次の例では、同じ段落で発生している 3 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 H1 内で、検索語 index が発生してい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。詳細については、89 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。

```
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。

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。たとえば、

```
("author", "pullitzer") & NOT "book"
```

用語 “book” を含むすべてのテキスト文書が、“author” または “pullitzer” の検索から除外されます。

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。

たとえば、

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

この検索は、つづりを誤ったワード Pulitzer の出現を検出できます。

突き合わせレベル (例では “80”) は、正確度を指定します。光学式文字認識装置または音声入力を使用して作成した文書の場合のように、文書につづりの誤りがある可能性が高い場合には、ファジー検索を使用してください。レベルとして 1 ~ 100 を使用します。100 は正確な一致を表し、80 はファジーを表します。

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”を検索するときに、“リポジトリ”と“DB2”のような用語も検出できます。

このタイプの検索は、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を対象にします。

シソーラスの作成方法の詳細については、91 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。データベースのサブジェクトに関する次の例は、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、“object relational database management system” という用語を指定し拡張することによって、シソーラス“ nsesamplethes” 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、“DB2” が追加され、検索されます。

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

次の例は、用語“ product” を指定し拡張することによって、そのすべての同義語 が追加されます。この場合は、“library” という同義語が 1 つあります。

```
SELECT CATEGORY, DATE
      FROM DB2EXT.HTMLTAB
      WHERE CONTAINS(HTMLFILE,
                    'THESAURUS "nsesamplethes"
                    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
```

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・テキスト照会でワードのシーケンスは問題になりませんが、照会用語のセットのうちの少なくとも 1 つの用語が、検索対象の文書で発生している必要があります。

文字またはワードのマスキングは、フリー・テキスト引き数の検索ストリングについては、サポートされません。

たとえば、

```
SELECT AUTHOR, TITLE
FROM DB2EXT.TEXTTAB
WHERE CONTAINS(COMMENT,
'IS ABOUT EN_US "something related to dinosaur"') = 1
```

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。

このファイルを実行するには、次のコマンドを入力します。

```
db2 -t vf search
```

表が作成済みでない場合には、db2ext/samples/ ディレクトリーで nsesample (.bat) を実行します。

ストアード・プロシージャー検索を使用するテキスト検索

ストアード・プロシージャー検索インターフェースは、ユーザーのアプリケーションがテキスト検索結果のサブセットを、高いハイパフォーマンスで、必要とする場合に使用されます。すべての結果が必要な場合、または大量の文書の索引が必要な場合に使用すべきではありません。理由としては、ユーザー表のある部分がメモリーにコピーされること、したがって、大量のメモリーが必要となることが挙げられます。

ストアード・プロシージャーでは、先ず 0 ~ 20 の結果を要求し、次に 21 ~ 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。カーソルの機能性と、キャッシュ (索引作成中に計算される) の使用、および索引作成中に結合の必要がないことから、検索は極めて高速です。

ストアード・プロシージャーを使用する場合は、以下の点を確認してください。

- CREATE INDEX の間に、Cache-search-result-options 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用メモリー所要量が、完全に考慮に入れられている。詳細は、38 ページの『ストアード・プロシージャー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 を使用して、索引のキャッシュが活動化されている。

ストアード・プロシージャー検索は以下ようになります。

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

その他の検索構文の例

ここで、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。最後の 2 つの整数値は、スコアが必要かどうか、および検索語が索引で検出される頻度の情報も求めるかどうかを指定します。

パラメーターの詳細については、161 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。

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が、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がある場合に、SQL 表値関数を使用してください。

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が、結果をユーザー表に結合する必要が依然としてあります。

たとえば、

```
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 total

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

--> numberofmatches  INTEGER
--> the hitcount in this document
```

単一の主キー列だけが許されてい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、157 ページの『DB2EXT.TEXTSEARCH』を参照してください。

パフォーマンスに関する考慮事項

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する DB2 オプティマイザーの実行プランをチェックしてください。
 - パラメトリック検索が、検索をより速くする可能性があります。他の検索の場合は、結果サイズを削減してください。
 - すべての結果を必要とするのではない場合には、結果の制限を使用してください。
 - `VARCHAR` タイプではなく、タイム・スタンプや整数などの小さな主キー列を使用してください。
- ストアド・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。

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 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を含めることができます。

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ることに注意してください。121 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、165 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、225 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 テキスト検索エンジン 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限定された語い

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、145 ページの『第 14 章 検索引き数の構文』を参照してください。

検索アプリケーションで使用するためにシソーラスを作成するには、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、テキスト検索エンジン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。テキスト検索エンジンは、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。

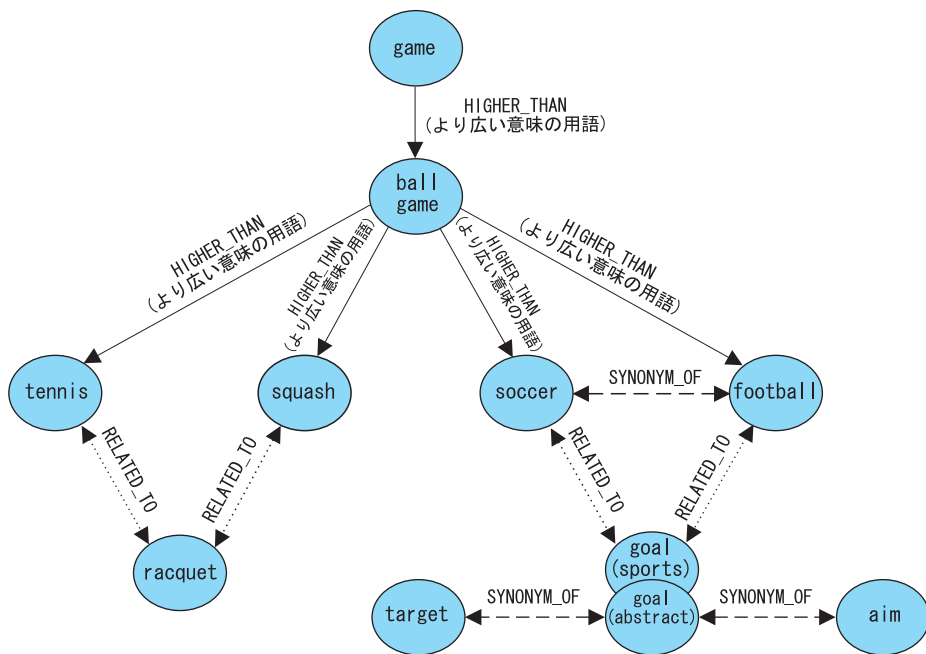


図 28. シソーラスの構造の例

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。HIGHER_THAN 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方向のものがあります。たとえば、HIGHER_THAN は単方向関係の名前です。

定義済みシソーラス関係

テキスト検索エンジン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
German RELATED_TO sausage
```

- 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
```

92 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:

– LOWER_THAN。幅が狭くなる関係をモデル化します。

LOWER_THAN 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、LOWER_THAN 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。

– HIGHER_THAN。幅の広くなる関係をモデル化します。

HIGHER_THAN 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くます。たとえば、HIGHER_THAN 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、LOWER_THAN、および HIGHER_THAN のシソーラス関係を定義できます。各関係名は固有なものでなければならぬため、RELATED_TO(42) などのように固有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。

LOWER_THAN(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

テキスト検索エンジン 関数で使用できるシソーラスを作成するには、以下のようになります。

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。

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 `thessamp.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。

```
drive:\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。

```
DB2_INSTOWNER/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
  administration
  .RELATED_TO management method
  .RELATED_TO personnel management

: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、177 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 164 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 ~ 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、137 ページの『DB2EXTTH (ユーティリティ)』を参照してください。

ヒント

249 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。

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

本章では、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、サービスのロックングおよび更新、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 Services 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。	100
START	DB2 Net Search Extender インスタンス・サービスを開始する。	102
STOP	DB2 Net Search Extender インスタンス・サービスを停止する。	103

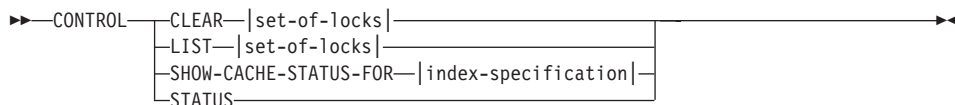
CONTROL

このコマンドを使用すれば、Net Search Extender Instance Services 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。ロックング・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、および活動化されたキャッシュに関する情報を表示できます。

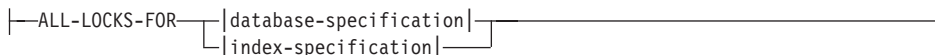
許可

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。

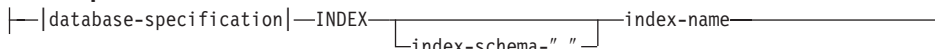
コマンド構文



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。ロックング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、より詳細な情報を得るには、LIST を使用します。

set-of-locks

指定されたデータベースまたは索引でのみロックを処理します。

SHOW CACHE STATUS FOR

指定された索引のキャッシュ表の活動化状況を表示します。状況は、「活動化されていない (Not Activated)」または「現在活動化されている (Currently Activated)」です。活動化されている場合は、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) などのキャッシュ・メモリーの使用に関する詳細が表示されません。

STATUS

STATUS キーワードを使用すると、このコマンドは Net Search Extender Instance Services のロッキングおよび更新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・メッセージがロッキング問題の発生を示している場合、ALTER コマンドの試行中に UPDATE コマンドのような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。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、活動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 コマンド

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活動化しないので、ストアード・プロシージャによる検索が必要な場合は、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 の場合、コマンドはサービス `db2ext-<InstanceName>` を開始します。これは、通常の Window 方式を使用して開始することもできます。

フル・テキスト索引のロックングの場合、構成ファイルは要件に応じて変更できます。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 Instance Services のロッキングおよび更新を停止します。

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

```
▶▶—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活動化されている場合であっても、サービスを停止します。指定されない場合は、コマンドはこれらのケースで失敗します。

使用法

Net Search Extender Instance Services を停止すると、Net Search Extender の特定のコマンドをそれ以上使用できなくなります。サービスを再始動する際には、一時キャッシュを再び活動化する必要があります。

活動化されたキャッシュまたは実行中の Net Search Extender コマンドは、サービスの停止を許可しないことに注意してください。

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

本章では、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から構成されています。

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 コマンドだけは、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することに注意してください。

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。	106
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。	108
DB2EXTDL (ユーティリティ)	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。	110

ヒント

これらのコマンド用に使用される実行可能な db2text によって、環境変数 DB2DBDFT で指定されたデフォルトの データベース が暗黙接続されます。

ENABLE DATABASE

このコマンドによって、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、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 権限を持つ必要があります。

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、省略できます。

USER userid USING password

userid は、password とともにデータベースに接続するために使用されます。

使用法

このコマンドは、DB2 Net Search Extender によって使用される、接続されたデータベースを準備します。これは、DB2 Net Search Extender 索引をデータベース内の表 / 列上に作成できるようになる前の必須ステップです。

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 Text Information Extender 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 187 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・サブシステムへの変更
なし。

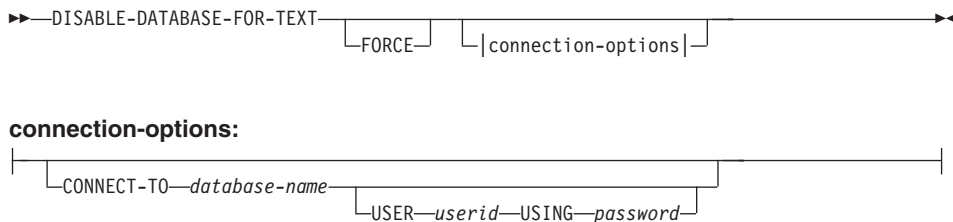
DISABLE DATABASE

このコマンドは、DB2 Net Search Extender がデータベースに加えた変更を元に戻します。

許可

データベースを使用不可にするには、このコマンドをデータベース管理者として、実行する必要があります。そのためには、DBADM 権限が必要です。

コマンド構文



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、省略できます。

USER userid USING password

userid は、password とともにデータベースに接続するために使用されます。

FORCE

データベース内のすべての DB2 Net Search Extender 索引を強制的にドロップさせます。索引をドロップさせる方法については 135 ページの『DROP INDEX』を参照してください。

使用法

このコマンドは、接続されたデータベースをリセットするため、他の DB2 Net Search Extender コマンドからそのデータベースを使用することはできなくなります。データベース内にフル・テキスト索引がある場合は、FORCE オプションを使用しない限り、このコマンドは失敗します。

DBADM 権限は DB2 インスタンス所有者から除去され「ない」ことに注意してください。

データベースへの変更

DB2 Net Search Extender を使用可能にするためにデータベースで行われた変更の大半は、取り消されます。下記の作業が含まれています。

- データベースでの DB2 Net Search Extender カタログ・ビューの削除。
- DB2 Net Search Extender によって作成されたすべてのデータベース・オブジェクトの削除。
- DB2 インスタンス所有者は DBADM 権限を維持する。

ファイル・システムおよび共用メモリーへの変更

FORCE オプションが使用されていれば、索引ファイルは削除されます。135 ページの『DROP INDEX』を参照してください。

索引の活動化されたキャッシュの場合、FORCE オプションが使用されていれば、キャッシュは削除されます。135 ページの『DROP INDEX』を参照してください。

DB2EXTDL (ユーティリティ) コマンド

DB2EXTDL (ユーティリティ)

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、100KB 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、この値は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 権限を持つ必要があります。

コマンド構文

```
▶▶DB2EXTDL—new-result-size—|connection-options|▶▶
```

connection-options:

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

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。

第 13 章 テキスト表所有者用の管理コマンド

本章では、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』では、これらのコマンドの使用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	キャッシュを活動化する。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	113
ALTER INDEX	索引の特性を変更する	115
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	119
CREATE INDEX	フル・テキスト索引を作成する	121
DEACTIVATE CACHE	キャッシュを非活動化する。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	133
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	137
DROP INDEX	テキスト列用のフル・テキスト索引をドロップする	135
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	106
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	139
HELP	DB2TEXT コマンド・オプションのリストを表示する	142
COPYRIGHT	Net Search Extender の製品および著作権情報を表示する	144

ヒント

これらのコマンド用に使用される実行可能な `db2text` によって、環境変数 `DB2DBDFT` で指定されたデフォルトのデータベースが暗黙接続されます。

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からキャッシュ済みの表を活性化します。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、161 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。

許可

DB2 カタログ・ビューに従って、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関して CONTROL 特権を持つ必要があります。

コマンド構文

```

▶▶ACTIVATE CACHE FOR INDEX index-schema."index-name"FOR-TEXT
    [RECREATE] [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せん。

index-name

CREATE INDEX コマンドで指定されたテキスト索引の名前。

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。活性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、ユーザーがサーバー上で、必要な DB2 権限を持つユーザー ID の下でコマンドを実行している場合は、省略できます。

ACTIVATE CACHE コマンド

USER userid USING password

userid は、password とともにデータベースに接続するために使用されます。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せん。

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、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

キャッシュ済み表の活動化は、永続キャッシュが使用された場合でも、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活動化されている間に、更新操作が実行された場合に発生します。

現行の文書数と結果の列のサイズを考慮に入れて、キャッシュの構築に必要なメモリーの量が動的に計算されます。CREATE コマンドまたは ALTER INDEX コマンドからの PCTFREE 値は、計算されたメモリーの最少量を増やすために $100/(100-PCTFREE)$ の係数で使用されます。

したがって、PCTFREE は、キャッシュが活動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。

さらにこのコマンドは、索引が CACHE TABLE オプションを指定して作成された場合のみ許可されることにも注意してください。詳しくは、121 ページの『CREATE INDEX』を参照してください。

ファイル・システムへの変更

必要に応じて、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。

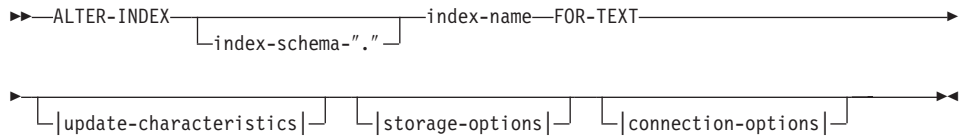
ALTER INDEX

コマンドは、更新および保管オプションのような、フル・テキスト索引の特性を変更します。

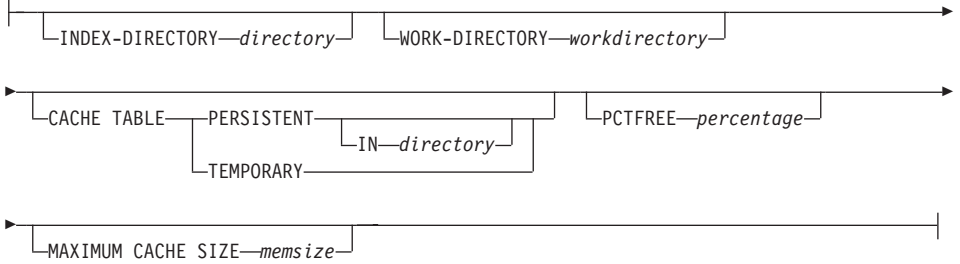
許可

DB2 カタログ・ビューに従って、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関して CONTROL 特権を持つ必要があります。

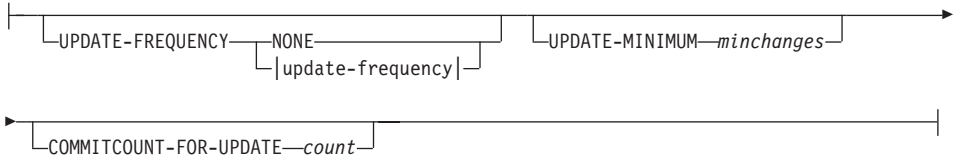
コマンド構文



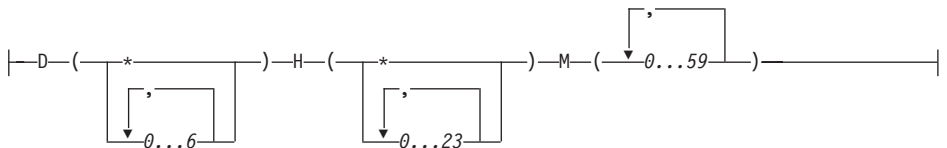
storage-options:



update-characteristics:

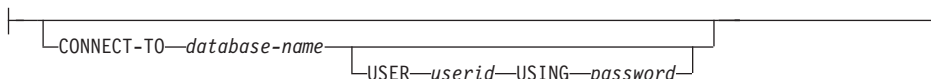


update-frequency:



connection-options:

ALTER INDEX コマンド



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。

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。

サブディレクトリー NODE0000 は、directory 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

検索および管理操作中に一時ファイルを保管するために使用されます。新規の索引ディレクトリーとは関係なく、別個の作業ディレクトリーを変更できません。

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。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。

サブディレクトリー NODE0000 は、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。

CACHE TABLE PERSISTENT IN directory

非活動化またはシステム・リポート後も CREATE INDEX で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 directory に保管されます。

以前に作成された永続キャッシュは、常に非活動化された索引が必要な新しいロケーションに移されます。

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、非活動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する必要があります。

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、非活動化された索引が必要です。

PCTFREE percentage

キャッシュのパーセントが追加の文書のために、フリーのまま保持されるように指定します。パーセントは、100 より小さく、0 以上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、この変更には非活動化された索引が必要であることに注意してください。詳細に関しては 113 ページの『ACTIVATE CACHE』を参照してください。

UPDATE FREQUENCY

索引更新の頻度は、更新が開始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.** 索引が更新される指定された日の時間 : * (毎時) または 0..23
- **M.** 索引が更新される指定された時間の分 : 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と、頻度設定は未変更のままです。

UPDATE MINIMUM minchanges

索引を徐々に増やして更新する前に、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、設定は未変更のままです。

索引が RECREATE ON UPDATE オプションを使用して作成されなかった場合にのみ、UPDATE MINIMUM を変更できる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。詳細に関しては 139 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。

索引が RECREATE ON UPDATE オプションを使用して作成されなかった場合にのみ、COMMITCOUNT を変更できることに注意してください。

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

ALTER INDEX コマンド

れ、ユーザーがサーバー上で、必要な DB2 権限を持つユーザー ID の下でコマンドを実行している場合は、省略できます。

USER userid USING password

userid は、password とともにデータベースに接続するために使用されます。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ず。

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

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 NODE0000 サブディレクトリーを作成します。

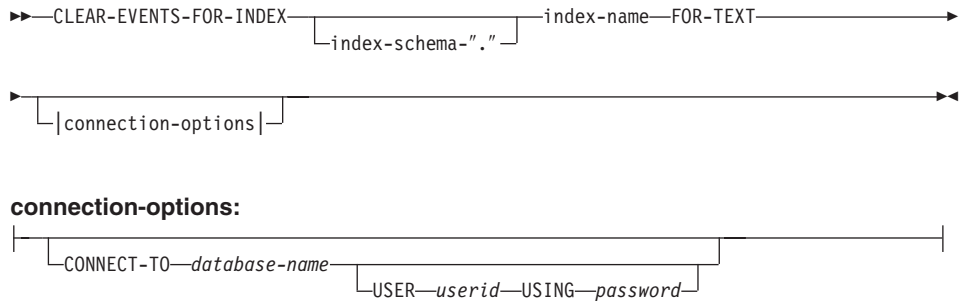
CLEAR EVENTS

このコマンドは、索引のイベント・ビューから索引付けイベントを削除します。このイベント・ビューは管理用に使用され、その名前は、DB2EXT.TEXTINDEXES ビューの EVENTVIEWNAME 列にあります。

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関して CONTROL 特権を持つ必要があります。

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、ユーザーがサーバー上で、必要な DB2 権限を持つユーザー ID の下でコマンドを実行している場合は、省略できます。

USER userid USING password

userid は、password とともにデータベースに接続するために使用されます。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

定期的な更新がスケジュールされている場合 (CREATE コマンドまたは ALTER INDEX コマンドの UPDATE FREQUENCY オプションを参照) は、イベント表を定期的にチェツ

CLEAR EVENTS コマンド

クする必要があります。索引の DB2 Net Search Extender イベント表をクリーンアップするには、イベントの理由を確認し、エラーの原因を除去した後で、CLEAR EVENTS を使用します。

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。

また、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

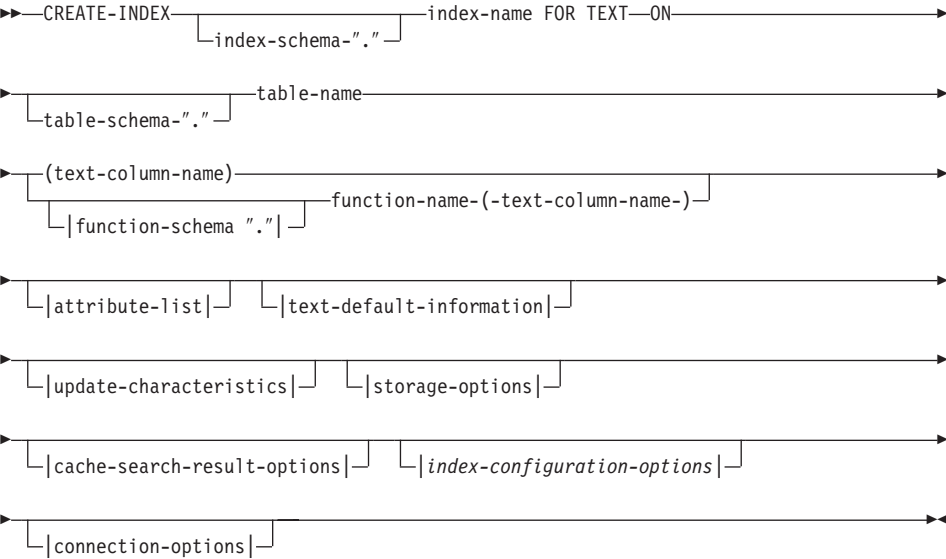
CREATE INDEX

このコマンドは、DB2 Net Search Extender フル・テキスト照会を使用するために、テキスト列上にフル・テキスト索引を作成します。

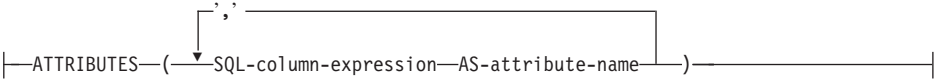
許可

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関して CONTROL 特権を持つ必要があります。

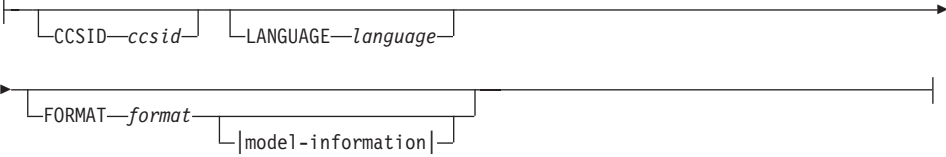
コマンド構文



attribute-list:



text-default-information:



model-information:

CREATE INDEX コマンド

|DOCUMENTMODEL—*documentmodel-name*—IN—*modelfilepath*—|
|USING-CCSID—*ccsid*—|

update-characteristics:

|UPDATE-FREQUENCY—NONE—|
|update-frequency—|
|incremental-update-characteristics—|
|RECREATE INDEX ON UPDATE—|

incremental-update-characteristics:

|UPDATE-MINIMUM—*minchanges*—| REORGANIZE—|
|AUTOMATIC—|
|MANUAL—|
|COMMITCOUNT-FOR-UPDATE—*count*—|

update-frequency:

|D—(—*—) —H—(—*—) —M—(—0...59—)
|0...6—| |0...23—|

storage-options:

|INDEX-DIRECTORY—*directory*—| |WORK-DIRECTORY—*workdirectory*—|
|ADMINISTRATION-TABLES-IN—*tablespace-name*—|

cache-search-results-options:

|CACHE TABLE—(—SQL-column-expression—)
|PERSISTENT—|
|IN—*directory*—|
|TEMPORARY—|
|PCTFREE—*percentage*—| MAXIMUM CACHE SIZE—*memsize*—|

┌──INITIAL SEARCH RESULT ORDER──(─SQL-order-by-list─)┐

┌──KEY COLUMNS FOR INDEX ON VIEWS──(SQL-columnname-list)┐

index-configuration-options:

┌──INDEX CONFIGURATION──(─option-value─)┐

connection-options:

┌──CONNECT-TO─database-name─┐
└──USER─userid─USING─password─┘

コマンド・パラメーター

index schema

テキスト索引のスキーマ。索引独自の管理表の DB2 スキーマ名として使用されます。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 がスキーマ名として使用されます。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、データベースのフル・テキスト索引を一意的に識別します。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 187 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 がスキーマ名として使用されます。

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。

CREATE INDEX コマンド

- ビューは、ストアード・プロシージャ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欄を指定する必要があります。
- 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が、ニックネーム表またはビューでデータへの変更が発生した場合、手作業で埋める必要があります。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は、第 13 章『ビューの説明』に記載されています。
- DB2 の述部 `contains/score/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。
- ビュー上の索引は、`cache-search-result-options` がコマンドで指定された場合にのみ許可されます。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。

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。この列は、次のタイプでなければなりません。

- 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 を指定して、トランスフォーメーション機能を使用し、列タイプを変換します。

データ・リンク列が使用されている場合は、参照された内容は、データ・リンク値の一部（たとえば、Http）であるプロトコルを介して、索引付けのためにフェッチされることに注意してください。“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・サーバーが必要なので、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に限られることに注意してください。

- 索引がビュー上に作成された。したがって、索引は CONTAINS、SCORE または NUMBER OF DOCUMENTS 検索引き数では使用できません。
- 索引が表上に作成された。この場合、CONTAINS、SCORE または NUMBER OF DOCUMENTS 引き数によって、どの索引が選択されるかは問題ではありません (3 つがすべて同期化されていれば)。これによって、次の CREATE INDEX コマンド詳細における同一の列上のすべての索引に対して、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

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、Net Search Extender によって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、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式に追加して索引付けされ、検索ステートメントの ATTRIBUTE 文節によって検索できることを決定します。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、索引のモデル定義ファイルの属性名とは異なる必要があります。

式の属性名は、以下の規則を使用して決まります。

- 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。たとえば: 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。たとえば: 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。

たとえば: ATTRIBUTES (CAST(JULIAN_DAY(date) AS DOUBLE) as day, (price1+price2)/2 as avg_price)

CREATE INDEX コマンド

引用符のない属性は英大文字にマップされ、検索時にはこのように指定する必要があります。

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'`。

LANGUAGE `language`

文書を索引付けする際に、文の終わりと段落の終わりの検出を判別する言語。サポートされる言語に関しては、201 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'`。

FORMAT `format`

列内の HTML などのテキスト文書のフォーマット。DB2 Net Search Extender は、文書を索引付けする際に、この情報を必要とします。構造化文書用にサポートされた文書フォーマットは、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』にリストされています。

構造化文書フォーマットの場合、文書モデル・ファイルで構造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。165 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'`。

DOCUMENTMODEL `documentmodel-name` IN `modelfilepath`

`modelfilepath` は、FORMAT 文節のフォーマット用のモデル定義を含むモデル・ファイルのロケーションを指定します。これは、DB2 インスタンス所有者が読めるものでなければなりません。文書モデルを使用すれば、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が、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。

文書モデルは、検索条件で参照する必要はありません。代わりにモデル・ファイルの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関しては、89 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 コマンド実行時にも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。

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'。

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.** 索引が更新される指定された日の時間: * (毎時) または 0..23
- **M.** 索引が更新される指定された時間の分: 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'。

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'。

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、変更の数がログ表およびトリガーがなければ利用できないので、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。

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は、索引のみを認識します。このステップは、更新後、select REORG SUGGESTED from DB2EXT.TEXTINDEXES の値に従って自動的に完了します。

REORGANIZE MANUAL は、REORGANIZE オプションを指定して、手動の UPDATE コマンドでのみ実行することができます。

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'。

REORGANIZE オプションの詳細に関しては、139 ページの『UPDATE INDEX』を参照してください。

CREATE INDEX コマンド

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 139 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'。

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。

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 139 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 NODE0000 は、サーバーの論理ノード上の索引を区別するために、directory の下に作成されます。

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'。サブディレクトリー NODE0000 は、サーバーの論理ノード上の索引を区別するために、directory の下に作成されます。

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

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選択されません。

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、フルテキストの検索結果を 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結果セットを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。たとえば: `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。たとえば: `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。

注

結果セットの列名が分離していない場合は、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、暗黙的には活動化されません。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。

PERSISTENT IN directory

キャッシュも永続的に作成され、非活動化またはシステム・リブートの直後に活動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 directory に保管されます。

ディレクトリー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='CACHEDIRECTORY'`。

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'`。

CREATE INDEX コマンド

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。

PCTFREE percentage

キャッシュのパーセントが追加の文書のために、フリーのまま保持されるように指定します。パーセントは、100 より小さく、0 以上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'。

詳細に関しては 113 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、文書は、キャッシュ結果表に保管された状態の索引付けされた順序で戻されます。

詳細については 161 ページの『第 16 章 ストアード・プロシージャ検索関数』をご覧ください。

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、指定しては「なりません」。列名のリストは、ビュー内の行を「一意的に」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	0 または 1	数字のシーケンスが文字に隣接していても、別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	0 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。

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、ユーザーがサーバー上で、必要な DB2 権限を持つユーザー ID の下でコマンドを実行している場合は、省略できます。

USER userid USING password

userid は、password とともにデータベースに接続するために使用されます。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ず。

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、DB2 索引に従った場合に限りません。
- 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: ユーザー・テキスト表上のトリガー作成を据え置きます。これは、RECREATE INDEX オプションが指定されていない場合に限りません。

共用メモリーへの変更

ACTIVATE の実行を据え置きます。CACHE TABLE 文節が使用された場合、結果表のキャッシュは、共用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 NODE0000 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>/NODE0000 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、NODE0000 は、分散 DB2 環境のノード番号に関係付けられます。

CREATE INDEX コマンド

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 version 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。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。

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。詳細に関しては 139 ページの『UPDATE INDEX』を参照してください。

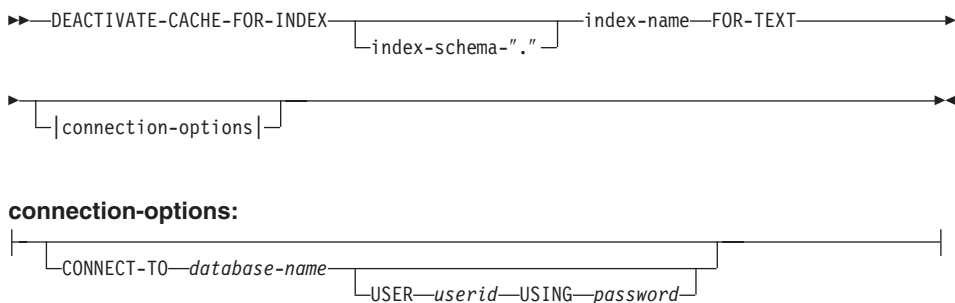
DEACTIVATE CACHE

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。次の活動化まで、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関して **CONTROL** 特権を持つ必要があります。

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、ユーザーがサーバー上で、必要な DB2 権限を持つユーザー ID の下でコマンドを実行している場合は、省略できます。

USER userid USING password

userid は、password とともにデータベースに接続するために使用されます。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。

DEACTIVATE CACHE コマンド

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

注

永続キャッシュの非活動化後、ストアド・プロシージャ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、最初の ACTIVATE の場合、その間に更新が実行されない限り使用できます。

この場合、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 コマンド

使用法

キャッシュ表の活動化状況に無関係に索引は削除されます。詳細については、113 ページの『ACTIVATE CACHE』を参照してください。

次のコマンドのいずれかが索引上で実行中の場合には、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。

共用メモリーへの変更

キャッシュ表が削除されます。

ファイル・サブシステムへの変更

- 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—codepage— -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 lkn。ただし、n は桁数。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 codepage

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、247 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, or -?

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 Text Information Extender V7.2 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、Text Information Extender 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、91 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

索引が最新の状態に更新されて、索引に関連付けられたテキスト列に現在含まれているデータが反映されることによって、このコマンドは、即時に索引付け処理を開始します。

更新を実行する一方で、CONTAINS 述部を使用した検索が可能です。活動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・プロシージャー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。

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関して CONTROL 特権を持つ必要があります。

コマンド構文

```

▶▶ UPDATE-INDEX [index-schema-"."] index-name FOR-TEXT [REORGANIZE]
▶ [COMMITCOUNT count] [USING-UPDATE-MINIMUM] [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。

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。

UPDATE INDEX コマンド

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。

詳細については、121 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、ユーザーがサーバー上で、必要な DB2 権限を持つユーザー ID の下でコマンドを実行している場合は、省略できます。

USER userid USING password

userid は、password とともにデータベースに接続するために使用されます。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ず。

使用法

このコマンドは、同期的に実行されます。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 187 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。

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 (NUMBERDOCS) 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・ビューから検索できます。

ただし、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

非活動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。

このコマンドがユーザーによって割り込まれた場合、更新機能に関する処理は、すべて停止されます。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。

注

このコマンドがユーザーによって割り込まれた場合、作成および初期更新機能に関する処理は、すべて停止されます。

コマンドは、すべての区画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

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。たとえば、次のように使います。

```
DB2TEXT ? CTE0110
```

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。これは、1 つまたは複数の検索項目と検索パラメーターから構成されています。

検索引き数の例は、80 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。85 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。

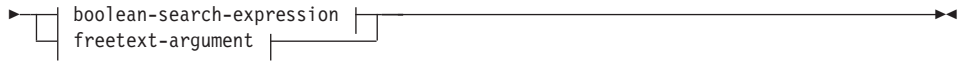
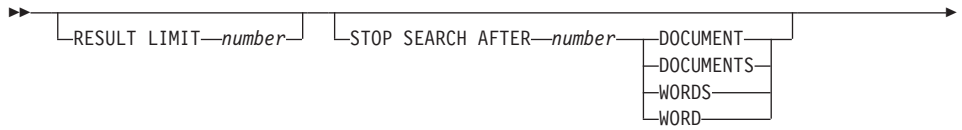
- **CONTAINS**。この関数は、特定のテキスト文書でテキストを検索するために検索引き数を使用します。文書にテキストまたは検索引き数に指定された関係が含まれている場合は、INTEGER 値 1 を返します。それ以外の場合は、0 を返します。
- **NUMBEROFMATCHES**。この関数は、テキスト文書内で検索するために検索引き数を使用し、文書ごとに一致した結果の数を示す INTEGER 値を返します。
- **SCORE**。この関数は、テキスト文書内で検索するために検索引き数を使用します。検出された文書ごとに値を返し、検出された文書が検索引き数によってどの程度適格に記述されているかを示します。

注

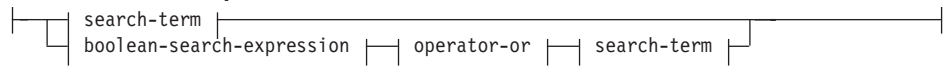
同じ構文がストアド・プロシージャ検索および SQL 表値関数の検索引き数で使用されています。

検索引き数

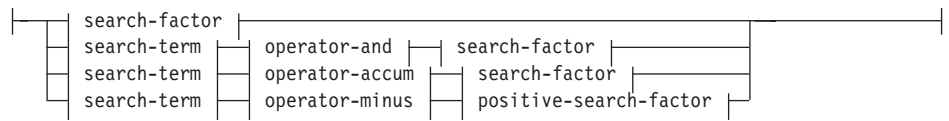
検索引き数の構文



Boolean-search-expression:



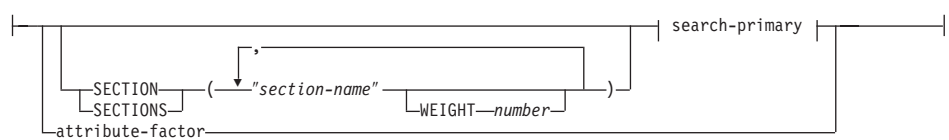
search-term:



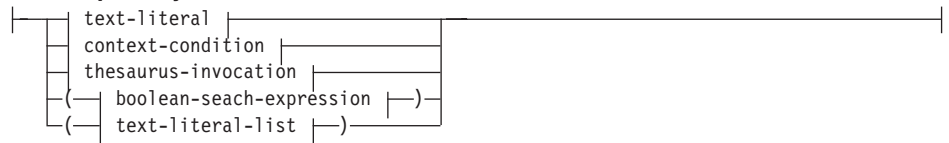
Search-factor:



Positive-search-factor:



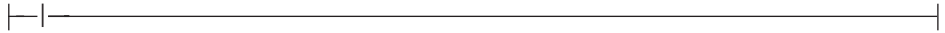
Search-primary:



Operator-and:



Operator-or:



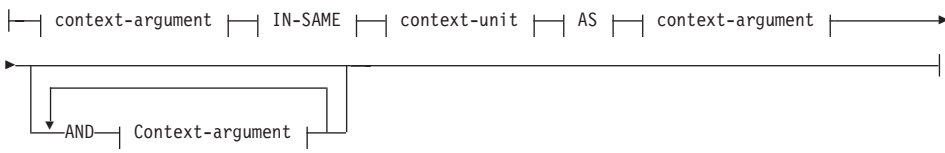
Operator-accum:



Operator-minus:



Context-condition:



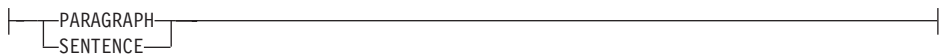
Context-argument:



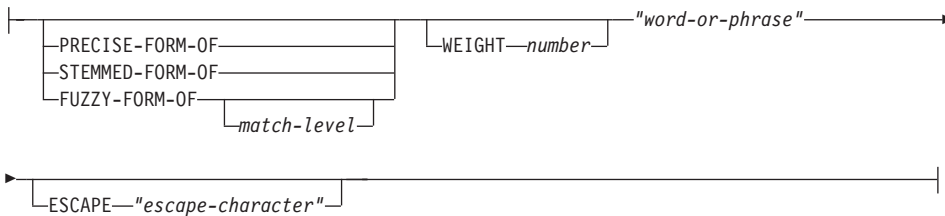
Text-literal-list:



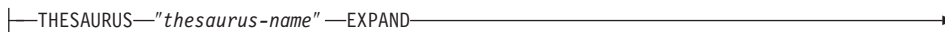
Context-unit:



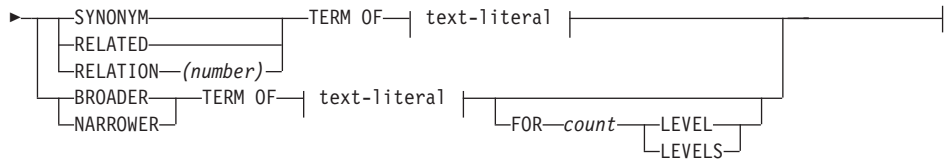
Text-literal:



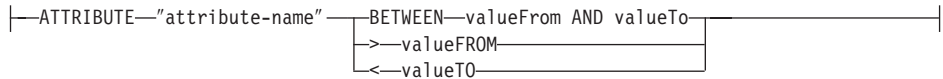
thesaurus-invocation:



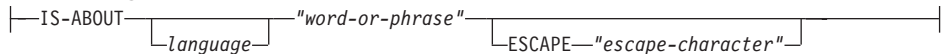
検索引き数の構文



Attribute-factor:



freetext-argument:



例

例は、80 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。

STOP SEARCH AFTER number DOCUMENTS(S)/WORDS(S)

検索しきい値を指定するキーワード。検索中に文書またはワードの出現数がこの値に達すると、検索が停止され、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

同じバインディング (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。121 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ため、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 をそのようなリ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 (paragraph resp. sentence) 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 (または複合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 (または複合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 econony 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、および検索アトム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、すでに非常に「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格になるので、検索の経過時間が長くなります。デフォルトの match level は 70 です。

WEIGHT 数値

text-literal を重み値に関連付けてデフォルトのランキングを変更します。許可された重み値は、0 ~ 1000 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ブランク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"
```

マスク文字

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、非英数字の後に続けることはできません。

ESCAPE エスケープ文字

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。

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せん。

>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と、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・アルゴリズムを取得するために使用する必要があります。つまり、用語アルゴリズムの頻度は、スコア値を戻すためではなく、文書内での用語の位置をチェックするために使用されます。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、文書に含まれる用語の数は多く、より高いスコア値が戻されます。

索引で 1 つの語が検出されるとすぐに、スコアのない使用が真であると評価され、language はフリー・テキスト引き数が書き込まれた言語を指定します。

language に許可される値は、201 ページの『付録 E. サポートされている言語』に記載されています。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。

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、これらの 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	154
NUMBEROFMATCHES	検索して、検出された一致の数を戻す	155
SCORE	検索して、検出されたテキスト文書のスコア値を戻す	156
DB2EXT.TEXTSEARCH	検出された主キーの表、一致の数またはスコア値 (あるいはその両方) を戻す SQL 表値関数	157

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、77 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。145 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許可されていません。

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。145 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、許可されていません。

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説明されているかを示すスコア値を、検出された文書ごとに返すことができます。

SCORE は、DOUBLE 値を返します。検索項目がより頻繁に文書で出現するほど、文書のスコアは高くなります。

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。145 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許可されていません。

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 SQL 表値関数を備えています。

関数の構文

```
db2ext.textSearch
(
  Query           VARCHAR(4096),
  IndexSchema     VARCHAR(128),
  IndexName       VARCHAR(128),
  resultFirstRow  INTEGER,
  ResultNumberRows  INTEGER,
  PrimKeyBinding  <supported types>, // same type as above
)

return table
(
  primKey          <supported types>, // same type as above
  numberOfMatches  INTEGER,
  score            DOUBLE,
  totalNbResults  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 詳細については、145 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。詳細については、121 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、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。値は 0 以上でなければなりません。

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。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、検索は、primaryKeyBinding が関連付けられる文書に制限されます。

たとえば、primaryKeyBinding を CAST(5 as BIGINT) に設定する場合、検索は、BIGINT 主キー値の「5」をもった単一の文書に制限されます。

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、単一列の主キーのみであることに注意してください。

関数パラメーター

以下の出力パラメーターは、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。

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。各行は同じ値をもつことに注意してください。

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、例外として共用メモリーが不要なので、索引を活動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。

複数列の主キー表上で作業するには、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。たとえば、

```
select s.id from
db2ext.sample s, table (db2ext.textSearch(
    "characteristics",
    'DB2EXT',
    'COMMANDS',
    1,
    20,
    cast(NULL as INTEGER))) t
where s.id = t.primkey;
```

第 16 章 ストアード・プロシージャー検索関数

Net Search Extender は、事前定義の結果表を戻すストアード・プロシージャー検索を備えています。結果表は、索引の作成中にキャッシュ表セクションで指定されます。ストアード・プロシージャー検索は、少数の結果を特定の順序で戻す必要がある場合に使用してください。

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。

DB2EXT.TEXTSEARCH (ストアド・プロシージャー検索用)

関数の構文

```
CREATE PROCEDURE db2ext.TextSearch(

    IN      query          VARCHAR(4096),
    IN      indexSchema   VARCHAR(128),
    IN      indexName     VARCHAR(128),
    IN      resultFirstRow INTEGER,
    IN      resultNumberRows INTEGER,
    IN      rankingFlag   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 詳細に関しては 145 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用。詳細に関しては 121 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・プロシージャー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 1 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・プロシージャー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フラグ・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。たとえば、次のようにします。

```
db2 create user temporary tablespace tempts managed by system
using /work/tempts.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。スコアが要求されると、スコア値を表示した追加行が戻されます。

関数パラメーター

入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りリストとして戻されます。

パラメーターに入力値として任意の NULL 以外の値が含まれていた場合にのみ、出力が計算されることに注意してください。

関数パラメーター

出力パラメーターを以下に示します。

totalNumberOfResults

照会結果リストで検出された結果の合計数。

使用法

ストアド・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 RESULT COLUMNS 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、スコア値が入っています。

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

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。

文書モデルの要素は、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lementも、他のどのElementに組み込まれているかに基づいて区別できます。
- GPP (general purpose parser) 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。

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、225 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。

デフォルトの文書モデル

HTML および XML 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を備えています。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、

文書モデル

- 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 の場合、各フィールドは対応するタグの名前を割り当てられます。
 - XML の場合、生成されるフィールド名は、完全なタグ・パスです。たとえば `/play/role/name`。
- 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 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

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 : <code><a></code> <code><address></code> <code><au></code> <code><author></code> <code><h1></code> <code><h2></code> <code><h3></code> <code><h4></code> <code><h5></code> <code><h6></code> <code><title></code> フィールド名は、タグ名。たとえば "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、XPath 表記のタグ・パス名。たとえば <code>"/play/title"</code> 。
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。

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を処理しますが、索引付けと検索の効率を上げるには、自分自身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。つまり、制約のないテキスト検索には、そのテキストの検索が含まれます。

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 (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tring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String。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ができます。
- 開始タグは別のタグのユニーク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Stringであってはなりません。

文書タイプ定義については、225 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、228 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Stringの 1 つを検索して文書をスキャンします。開始境界Stringを検出すると、対応する終了境界String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Stringと終了境界String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。たとえばフォント情報など。
- 定位置情報を提供するタグ。たとえば新規の段落用の `<p>`; など。
- 構造上の情報を提供するタグ。たとえば `<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ー参照 (たとえば `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                                - This is the start of text field
```

```
name="header1"
tag="h1"
exclude="YES" /> - This is the end of the text field

<HTMLAttributeDefinition
name="year"
tag="meta"
meta-qualifier="year"
type="NUMBER" /> - This is the start of the document
attribute
- 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ています。

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もちます。

```
<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、225 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、228 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

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。

以下に XML 文書の例を示します。

```
<?xml version="1.0"?>
<purchaseOrder orderDate="2001-01-20">
  <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> [3]
    <price>239.90</price>
    <shipDate>2001-01-25</shipDate>
  </item>
  <item partNo="987" quantity="1"> [3]
    <name>Multifunction Rake ZYX</name>
```

文書モデル

```
<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"
    exclude="no" />

  <XMLFieldDefinition [2]
    name="customerName"
    locator="//shipAddress/name"
    exclude="yes"/>

  <XMLAttributeDefinition [3]
    name="partNumber"
    type="NUMBER"
    locator="/purchaseOrder//item/@partNo" />

</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、228 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LEMENT の実際のローカル名 (接頭部なし)。

\$(PATH)

XPath に一致する XML エLEMENT 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。

これらのロケーターは例からのものです。詳細については、226 ページの『ロケーター (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

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 : (NCName の後にコロンが続きます)、たとえば NCName:NCName

リテラルは、単一または二重引用符で囲まれるストリングです。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 ロケーターは、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。パラメーターを使用して、ロケーターに例外を指定します。

時には * などの汎用ロケーターを指定して、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ー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ー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、このロケーターに汎用ロケーター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 $-1 \sim +1$ 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。

この競合する解は、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。

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 7. テキスト索引の項目

フィールド名	索引付けされるテキスト	
addresses	123 Maple Street Mill Hill CA 90999	[1]
customerName	Alice Smith	[2]
partNumber	123 987	[3]

[1] では、shipAddress エレメントのテキストが、フィールド名 addresses の下で索引付けされ、name エレメント (Alice Smith) は索引付けされないことに注意してください。これは、name エレメント自体が文書モデル内において、フィールド名 customerName [2] の下で索引付けされるからです。つまり、組み込まれたエレメントは文書モデル内で許可されていますが、組み込まれたテキスト・フィールドの一部として索引付けされることはない、ことを意味しています。これらは別途、索引付けされません。

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または処理命令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・テキストまたは処理命令テキストです。
- 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、他の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。

文書モデル

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エンティティ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。

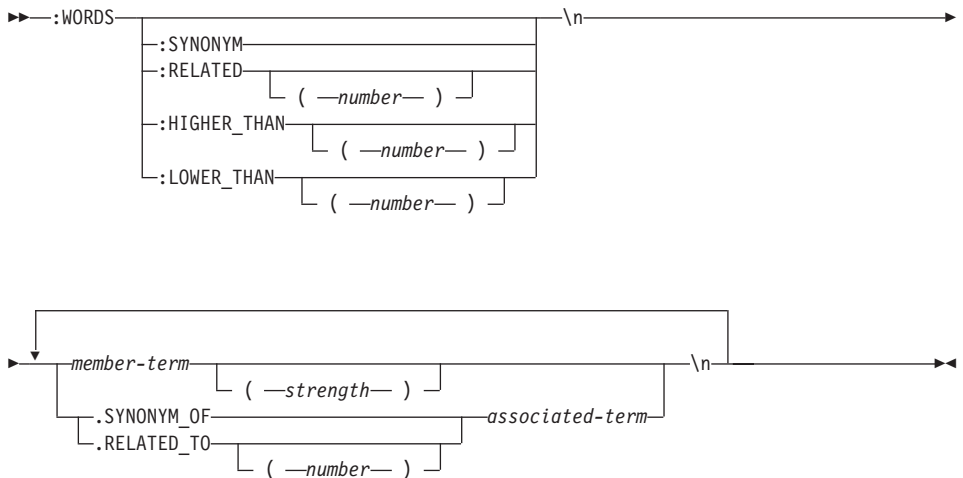
文書タイプ定義については、225 ページの『文書モデル用の DTD』を参照してください。

制約事項については、228 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。

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)],

:HIGHER_THAN [(number)],

:LOWER_THAN [(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。たとえば、

```
: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)]

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。たとえば、

```
: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、1 から 128 までの重複しない数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。たとえば、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。例を示します。

:WORDS:SYNONYM

reject

decline

RELATED_T0(1) accept

- 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。

第 3 部 付録

付録 A. マイグレーション

DB2 Net Search Extender バージョン 8.1 は、DB2 Text Information Extender 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 2 つのオプションが使用可能です。

- Net Search Extender バージョン 7.2 から Net Search Extender バージョン 8.1 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、古いインターフェースと互換インストール・パッケージを使用し続けることができます。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 へ自動的にマイグレーションする方法はありません。

- Text Information Extender バージョン 7.2 から Net Search Extender バージョン 8.1 への移動

Net Search Extender バージョン 8.1 をインストールし、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 にマイグレーションします。次に、db2extiupdt を使用してユーザーのインスタンスをマイグレーションしてから、db2extiupdt <dbname> を使用してデータベースをマイグレーションします。このプログラムは、すべての db2ext 表を読み取り、新しい構文を使用して、必要な索引のリスト <dbname>migto81.lst を生成します。

古い索引はすべてドロップされ、<dbname>migto81.lst に従って再作成されます。この処理には時間がかかりかかる場合があることに注意してください。

また、DB2 Text Information Extender バージョン 7.2 は、もう使用できないことにも注意してください。

注

最新のマイグレーション情報を入手するには、CD 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。

注

「”」文字は、Windows 上のデータベース・オブジェクトではサポートされていません。

付録 B. 大量のメモリーの使用

ストアド・プロシージャー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
- Sun Solaris

AIX

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 としてログインします。
 - 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。

共用メモリーしきい値の構成は以下のとおりです。

- AIX では、共用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 ユーティリティを使用して、大きい方の数値を選択します。

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。

Sun Solaris

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` としてログインします。
 - 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
```

共用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` を使用して共用メモリー・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。

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。この情報を表示するには、表上にいくつかのビューを照会できます。

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` 表ビュー

- 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 Text Information Extender 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。

これらのデータベース・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`

```
Select DEFAULTNAME, DEFAULTVALUE From DB2EXT.DBDEFAULTS
```

表 8. db2ext.dbdefaults ビュー

デフォルトの列名	デフォルト値	備考
CCSID	データベースの CCSID	文書のデフォルトの CCSID : CREATE INDEX コマンドで CCSID が指定されない場合に適用される。
FORMAT	TEXT	文書のデフォルト・フォーマット : CREATE INDEX コマンドでフォーマットが指定されない場合に適用される。
INDEXDIRECTORY	パス名を参照	フル・テキスト索引ファイルのディレクトリ : CREATE INDEX コマンドで索引ディレクトリが指定されない場合に適用される。 \$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	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。
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	パス名を参照	索引一時ファイル用のディレクトリ。 <os_dependent>/sql1lib/db2ext/indexes
CACHEDIRECTORY	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。<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 50。
USERPERSISTENTCACHE	1	永続キャッシュ。
AUTOMATICREORG	1	CREATE INDEX の REORGANIZE オプション。これは自動再編成を意味する。
TREATNUMBERSASWORDS	0	0 (CREATE コマンド、索引構成を参照)。

表 8. *db2ext.dbdefaults* ビュー (続き)

デフォルトの列名	デフォルト値	備考
INDEXSTOPWORDS	1	索引付け中にストップワードを無視する。

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 9. *db2ext.proxyinformation* ビュー

列名	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

索引レベルの情報は、次の DB2 Net Search Extender ビュー、または索引の UDF を使用して照会できます。

- *db2ext.textindexes*
- *db2ext.textindexformats*
- *db2ext.indexconfiguration*
- `<index eventview name schema>.<index eventview name>`

旧版との互換性の理由から、DB2 Text Information Extender 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。

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。

テキストの索引が作成されるたびに新規エントリーが db2ext.textindexes に作成されます。エントリーは、テキストの索引がドロップされると削除されます。

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。

```
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。

表 10. db2ext.textindexes ビュー

列名	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ー。
WORKDIRECTORY	VARCHAR(256)	索引一時ファイル用のディレクトリー。
CACHEDIRECTORY	VARCHAR(256)	永続キャッシュのディレクトリー (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。

表 10. db2ext.textindexes ビュー (続き)

列名	タイプ	備考
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。
EVENTVIEWNAME	VARCHAR(128)	この索引用に作成されるイベント・ビューの名前。
LOGVIEWSHEMA	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。
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。
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。
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。
VIEWKEYS	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.textindexformats 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です。次は、索引名を使用した例です。

```
Select VALUE from DB2EXT.INDEXCONFIGURATION where INDEXSCHEMA='myschema'
and INDEXNAME='myindex' and PARAMETER ="index stopwords"
```

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 11. db2ext.indexconfiguration ビュー

列名	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 VALUE	VARCHAR(30)	TREATNUMBERASWORDS 0 または 1 INDEXSTOPWORDS 0 または 1
FORMAT	VARCHAR(512)	モデルはこのフォーマットにバインドされています。

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。

```
Select FORMAT from DB2EXT.TEXTINDEXFORMATS where INDSCHEMA='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 12. db2ext.textindexformats ビュー

列名	タイプ	備考
INDSCHEMA	VARCHAR(128)	索引のスキーマ名 (ログ表での INDEXNAME 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。

表 12. db2ext.textindexformats ビュー (続き)

列名	タイプ	備考
MODELCCSID	INTEGER	MODELFILE の CCSID。
DEFAULT	INTEGER	現行は常に 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

索引レベルの情報は、次の DB2 Net Search Extender ビュー、または UDF を使用して照会できます。

- 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。ビューは、通常の SQL 照会機能を使用して、カタログ・ビューからイベント表のビューで使用できます。次は、この操作を使用した例です。

```
Select EVENTVIEW SCHEMA, EVENTVIEWNAME from DB2TEXTINDEXES
      where INDSHEMA = <mysystem> and INDNAME <INDNAME>
```

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 13. イベント・ビュー

列名	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)。
TIME	TIMESTAMP	イベント・エントリ作成のタイム・スタンプ。
REASON/SEVERITY	INTEGER	理由コード。
MESSAGE	VARCHAR(1024)	追加のテキスト情報。
KEY1, ... KEY14		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)。

索引レベル情報のビュー

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 119 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

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。

正規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。更新コマンドはエントリーを読み取り、正常な同期の後、削除します。

ビューに関する索引の場合に限り、トリガーはログ表に書きこむことができず、代わりにユーザーがこの作業の責任を負います。ビューは更新できます。

表 14. ログ表ビュー

列名	タイプ	備考
OPERATION	INTEGER	
TIME	TIMESTAMP	
PK01 ...		
PKnm		

OPERATION は、索引の同期 (0=挿入、1=更新、2=削除) を必要とするユーザー表上の操作です。TIME は、この表の行を作成するためのタイム・スタンプです。PK01、...,PKnm 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。

ユーザー表を定義すると、このビューを選択、更新、挿入、および削除する権限が付与されます。

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。

SBCS

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
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、キリル語、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	キリル語 - 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-bit 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
1025	キリル語、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	キリル語、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 ギリシャ語 マルチリンガル (ユーロ文字付)

SBCS

1155	EBCDIC トルコ語 (ユーロ文字付)
1156	EBCDIC バルト語 (ユーロ文字付)
1157	EBCDIC エストニア語 (ユーロ文字付)
1158	EBCDIC キリル語、ウクライナ語 (ユーロ文字付)
1159	中国語 (繁体字) EBCDIC
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)
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

DBCS

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)
939	日本語
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
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
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
5354	ベトナム語

UNICODE

1200	UCS2
1208	UTF8

UNICODE

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、有意な言語独自の処理は、タイ語で書かれた文書に対してのみ行われます。

EN_US	英語 / 米国
EN_GB	英語 / 英国
EN_ZA	英語 / 南アフリカ
EN_AU	英語 / オーストラリア
EN_BE	英語 / ベルギー
FR_FR	フランス語 / フランス
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
DE_DE	ドイツ語 / ドイツ
DE_CH	ドイツ語 / スイス
IT_IT	イタリア語 / イタリア
IT_CH	イタリア語 / スイス
PT_BR	ポルトガル語 / ブラジル
ES_ES	スペイン語 / スペイン
ZH_CN	中国語 / 中国
ZH_TW	中国語 / 台湾
JA_JP	日本語 / 日本
KO_KR	朝鮮語 / 韓国
AR_AA	アラビア語 / アラビア語圏
CS_CZ	チェコ語 / チェコ
NL_NL	オランダ語 / オランダ
NL_BE	オランダ語 / ベルギー
DA_DK	デンマーク語 / デンマーク
FI_FI	フィンランド語 / フィンランド

言語

EL_GR	ギリシャ語 / ギリシャ
HE_IL	ヘブライ語 / イスラエル
IW_IL	ヘブライ語 / イスラエル
HU_HU	ハンガリー語 / ハンガリー
NO_NO	ノルウェー語 / ノルウェー
NB_NO	ノルウェー語ブークモール / ノルウェー語
NN_NO	ノルウェー語ニーノシュク / ノルウェー語
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
RU_RU	ロシア語 / ロシア
SV_SE	スウェーデン語 / スウェーデン
TR_TR	トルコ語 / トルコ
SQ_AL	アルバニア語 / アルバニア
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	カタルーニャ語 / スペイン
HR_HR	クロアチア語 / クロアチア
ET_EE	エストニア語 / エストニア
LV_LV	ラトビア語 / ラトビア
LT_LT	リトアニア語 / リトアニア
MK_MK	マケドニア語 / マケドニア
RO_RO	ルーマニア語 / ルーマニア
SR_SP	セルビア語 (キリル文字) / セルビア
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
TH_TH	タイ語 / タイ
UK_UA	ウクライナ語 / ウクライナ
VI_VN	ベトナム語 / ベトナム
ID_ID	インドネシア語 / インドネシア
HI_IN	ヒンディー語 / インド

MS_MY

マレー語 / マレーシア

TA_IN

タミル語 / インド

TE_IN

テルグ語 / インド

付録 F. Net Search Extender メッセージ

DB2 Net Search Extender には、以下のタイプのメッセージがあります。

- 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	更新およびロッキング・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	キャッシュの活動化が開始されました。
CTE0012	キャッシュの活動化が終了しました。
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可になりました。

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クング・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、索引を作成してみてください。

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。

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隔離ユーザー 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開けません。

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、実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クング・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が必要です。

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クング・サービスを再始動してください。

CTE0120 更新およびロックング・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クング・サービスを再始動してください。

CTE0121 更新およびロックング・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 更新およびロックング・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: 更新およびロックング・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: 更新およびロックング・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、共用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数を超えています。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バ

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

エラー・メッセージ

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

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。

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 つあります。

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、許容範囲内でなければなりません。

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せん。

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、操作を再実行してください。

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 ブランク文字はエスケープ文字として許可されません。

説明: エスケープ文節にブランク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・ディレクトリー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か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 共用システム・リソースの使用でエラーが発生しました。

説明: 共用メモリーまたはセマフォのような共用システム・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、共用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す。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す。

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)。

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「一意的に」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順序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こと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エントリー "[%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" で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: 処置は必要ありません。

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。

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。

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。これが必ず行な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せん。

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ため注意してください。 **DISABLE DATABASE** コマンドの場合、そのデータベース上のその他の

エラー・メッセージ

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す。

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・テーブルは、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・テーブルをチェックし、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、誤った形の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" を超過しています。

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、作業ディレクトリー、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。キャッシュを活動化 (DB2TEXT ACTIVATE コマンド) する際に、あるいはキャッシュが活動化されているとき

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にかき換えてください。

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。

エラー・メッセージ

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。されていないければ、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、索引がすでに活動化されています。

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 共用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、共用メモリー・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、共用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 共用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、共用メモリー・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 グローバル共用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、グローバル共用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 グローバル共用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、グローバル共用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活動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にのみ、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活動化されている索引のキャッシュを非活動化するか、**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: 索引が増分方式で更新される場合のみ、**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 索引の活動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: 索引の活動化処理中に、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。

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、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。

付録 G. 文書モデルの参照情報

- 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

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` に一致する祖先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` は付録祖先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::* (or @*)` 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 命名推奨で定義されたとおりです。

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。

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

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切断して、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、空白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。改行および空白文字などの制御文字は、次のように解釈されます。行の間にある制御文字 (0x20 より小) は、空白文字と見なされます。改行 (0x0A) の前後にある空白文字と制御文字は、無視されます。1 バイト文字の前後の改行は空白文字と見なされ、同じ文字の 2 バイト文字は、常に同じ文字と見なされます。同じ文字の大文字と小文字、たとえば、"A" と "a" は、検索時になにも指定されていない場合には、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、空白文字が後に続く「.」、「!」、「?」、「.」、および行の末尾にある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です。通常、"and"、"or"、および "in" などの、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。

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
DE_CH	スイスで話されるドイツ語
EN_GB	英国で話される英語
FI_FI	フィンランドで話されるフィンランド語
HE_IL	イスラエルで話されるヘブライ語
IW_IL	イスラエルで話されるヘブライ語
NN_NO	ノルウェーで話されるノルウェー語ニーノシュク
RU_RU	ロシアで話されるロシア語
CA_ES	スペインで話されるカタロニア語
DE_DE	ドイツで話されるドイツ語
EN_US	米国で話される英語
FR_CA	カナダで話されるフランス語
IS_IS	アイスランドで話されるアイスランド語
NN_NO	ノルウェーで話されるノルウェー語ニーノシュク
NB_NO	ノルウェーで話されるノルウェー語ブークモール
PT_BR	ブラジルで話されるポルトガル語
SV_SE	スウェーデンで話されるスウェーデン語
DA_DK	デンマークで話されるデンマーク語
EL_GR	ギリシャで話されるギリシャ語
ES_ES	スペインで話されるスペイン語
FR_FR	フランスで話されるフランス語
IT_IT	イタリアで話されるイタリア語
NL_BE	ベルギーで話されるオランダ語
PT_PT	ポルトガルで話されるポルトガル語

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ー制限のため、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。

理由コード

- 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 (たとえば、予期しないヌル・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 GTR 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 GTR 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 o fhandle 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。
 - 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いと、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。

理由コード

- 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。
- 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、定義ファイルのエントリー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オペレーター値
- 364** 列挙型範囲外のランク公式の値。
- 365** プロキシシー・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。

理由コード

- 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 (検索項目など) がありません。
- 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・コンバーターは無効なディスクリプターをもっています。
- 600** XML 文書に非同期エンティティ、つまり引用符で囲まれていない XML 属性値が含まれています。
- 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; 囲み開始タグおよび終了タグの外にテキストがあります。
- 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。

理由コード

- 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」で文書を処理中にエラーが発生しました。

付録 J. トラブルシューティング

DB2 Net Search Extender は、以下のトラブルシューティング情報を提供します。

- 障害のトレース

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*」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。

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 15. 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。

表 15. データ・リンク警告メッセージ (続き)

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。
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すぎます。
01H14=	CTEDL - 要求されたデータ・リンク URI が長すぎます。
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。

表 15. データ・リンク警告メッセージ (続き)

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。

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)
970	結合された韓国語
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)
850	PC データ Latin 1
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
819	Latin 1
874	タイ語
5039	日本語 (結合された SBCS/DBCS)
954	日本語
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
1208	UTF 8

シソーラス定義ファイルをコンパイルするには、137 ページの『DB2EXTTH (ユーティリティー)』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。

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・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・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* でリレーシオ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* でリレーシオ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* でリレーシオ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。

シソーラス・ツールが戻すメッセージ

付録 N. 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、プログラムまたはサービスの操作性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。実施権の許諾については、下記の宛先に書面にてご照会ください。

〒106-0032 東京都港区六本木 3 丁目 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。

本書に対して、周期的に変更が行われ、これらの変更は、文書の次版に組み込まれます。IBM は、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。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すが、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。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にお願います。

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A

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。したがって 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・プログラムのすべての部分、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
DB2 エクステンダー	

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。

UNIX は、The Open Group がライセンスしている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などはそれぞれ各社の商標または登録商標です。

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・テーブルのビュー。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。たとえば、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。使用可能化された表の文書が追加、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]

一致

検索結果内 79

NUMBEROFMATCHES 関数 155

イベント・ビュー 193

インスタンス・サービス 9, 27, 99

インストール 13

インストール検査 16

インフォメーション・カタログ 187

エスケープ文字

使用法 82

[カ行]

外部保管データ 9

開始、DB2 Net Search Extender の
102

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 7

追加概念 9

ビュー 10

表値関数の使用 8

列変換関数 9

SQL スカラー検索関数の使用 6

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

関数

概要 153

検索関数 77

関数 (続き)

検索引き数の指定 80

参照 153

ストアード・プロシージャ
162

説明 77

データ・タイプの変換用 36

テキストの検索 78

CONTAINS 154

NUMBEROFMATCHES 155

SCORE 156

SQL 表値 157

管理

インスタンス所有者のコマンド・
サマリー 99

キャッシュの活動化 75

キャッシュの非活動化 75

索引イベントのクリア 74

索引イベントをクリアする 46

索引状況の表示 76

索引のバックアップとリストア
47

障害のトレース 241

データベース管理者コマンドのサ
マリー 105

テキスト索引状況を表示する 47

テキスト索引設定の変更 45, 69

テキスト索引の更新 43, 73

テキスト索引の作成 34, 53

テキスト索引のドロップ 46, 72

テキスト索引の保守 43, 68

テキスト表所有者のコマンド・サ
マリー 111

ロッキング・サービスの使用 28

DB2 Net Search Extender の開始
27, 50, 102

DB2 Net Search Extender の停止
27, 50, 103

DB2 コントロール・センター
49

「キャッシュの活動化 (Activate the
cache)」ダイアログ 75

「キャッシュ表」パネル 62

「キャッシュ表の非活動化
(Deactivate the cached table)」ダイ
アログ 75

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 201

検索関数

ストアード・プロシージャ
162

CONTAINS 154

NUMBEROFMATCHES 155

SCORE 156

SQL 表値 157

検索項目の拡張

参照：シソーラス

検索項目の出現 155

検索語でのマスク文字 81

検索語でのワイルドカード文字 81

検索引き数

同じ段落内で用語を検索する 82

同じ文または同じ段落内で用語を
検索する 82

構文 146

固定シーケンスでの用語検索 82

シソーラス検索 84

指定 80

数値属性検索 84

説明 145

属性名 152

任意のシーケンスでの用語検索
80, 82

ブール演算子 148

ファジー検索 83, 150

フリー・テキスト検索 84

文書のセクション内で用語を検索
する 83

マスク文字の使用 81

用語の一部を検索する 81

検索引き数 (続き)
ワイルドカード文字の使用 81
NOT を使用して検索する 83
search-primary 演算子 149
& と | を使用して検索する 80
検索引き数キーワード
COUNT 151
EXPAND 151
FUZZY FORM OF 150
PRECISE FORM OF 149
RESULT LIMIT 148
SECTION 149
STEMMED FORM OF 150
STOP SEARCH AFTER number
DOCUMENTS(S)/
WORDS(S) 148
TERM OF 151
THESAURUS 151
検索引き数の & (AND) 演算子
使用方法 80
検索引き数の | (OR) 演算子
使用方法 80
更新サービス 30
「更新特性」パネル 60
更新の頻度 44
構造化文書
概要 89
検索構文 149
セクション・サポートを可能にする 165
デフォルトの文書モデル 165
例 83
コマンド
ACTIVATE CACHE 113
ALTER INDEX 115
CLEAR EVENTS 119
CONTROL 100
COPYRIGHT 144
CREATE INDEX 121
DB2EXTDL 110
DB2EXTTH 137
db2text 99, 105, 112
DEACTIVATE CACHE 133
DISABLE DATABASE 108
DROP INDEX 135
ENABLE DATABASE 106

コマンド (続き)
HELP 142
START 102
STOP 103
UPDATE INDEX 139
コマンド・サマリー
インスタンス所有者のための 99
データベース管理者用の 105
テキスト表所有者のための 111

[サ行]

サーバー
始動 102
障害のトレース 241
停止 103
索引
概要 3
関係 92
キャッシュの活動化 75
キャッシュの非活動化 75
計画 23
更新の頻度 44
構造化文書の使用 89
サイズの計算 23
索引イベントのクリア 74
索引イベントをクリアする 46
索引状況の表示 76
テキスト索引状況を表示する 47
テキスト索引設定の変更 45, 69
テキスト索引の更新 43, 73
テキスト索引のドロップ 46, 72
テキスト索引の保守 43, 68, 91
バックアップおよびリストア 47
DB2 コントロール・センター
49
「索引イベント」ダイアログ 74
索引更新イベント
記録 35
削除 46
「索引状況 (Index Status)」ダイアログ 76
「索引の更新」ダイアログ 73
索引のスペース所要量 23
索引のディスク・スペース 23
「索引のドロップ」ダイアログ 72

「索引の変更 (Alter index)」ダイア
ログ 69
索引を再作成する 44
索引をバックアップおよびリストア
する 47
サマリー・パネル 67
サンプル関数
実行 85
システム要件 13
シソーラス
概念 91
構造 91
コンパイル 94
作成 94
サポートされている CCSID 247
シソーラス定義構文 177
定義ファイル 94
メッセージ 249
シソーラス検索
構文 151
例 84
THESAURUS キーワード 151
シソーラスでの関係 92
シソーラスの条件の深さ、指定 151
シソーラス・コンパイル・ユーティ
リティ 137
主キー・タイプ 132
主要な概念 3
主要な機能 10
主要な用語 3
障害の検出 241
障害のトレース 241
使用する、大量のメモリーを 185
ストアード・プロシージャ
概要 38
テキスト索引の活動化 40
テキスト索引の更新 40
テキスト索引の非活動化 40
ビューに基づくテキスト索引 41
ストアード・プロシージャ関数
検索 85
構文 162

[タ行]

「ターゲット (Target)」パネル 55

大量のメモリーの使用
AIX の場合 185
Sun Solaris の場合 186
Windows の場合 185

追加概念 9

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
47
データベースの使用可能化 32,
51
データベースの使用不可化 33,
51

データ・リンクの戻りサイズの変更
38, 51

停止、DB2 Net Search Extender の
103

ディレクトリーおよびファイル名
16

テキスト検索エンジン
ストップワード 231
ストップワードをサポートする言
語 232
トークン化 231
理由コード 233

テキスト特性
フォーマット 23
CCSID 23

テキストの検索
概要 78
検出した文書のスコアを入手する
79
構文 146
照会を行う 78
ストアド・プロシージャを使
用する 85
表値関数の使用 86
見つかった一致の数を入手する
79

テキスト文書のフォーマット 23
サポート対象のリスト 23
説明 23
「テキスト・プロパティ」パネル
58

デフォルトの文書モデル 165

トラブルシューティング 241

トリガー
作成 35
説明 5

[ナ行]

「名前」パネル 53

入門 17

認識
ストップワード 231
段落 231
文 231
ワード 231

[ハ行]

パネルおよびダイアログ
「キャッシュの活動化 (Activate
the cache)」ダイアログ 75
「キャッシュ表」パネル 62
「キャッシュ表の非活動化
(Deactivate the cached table)」ダ
イアログ 75
「更新特性」パネル 60
「索引イベント」ダイアログ 74
「索引状況 (Index Status)」ダイ
アログ 76
「索引の更新」ダイアログ 73
「索引のドロップ」ダイアログ
72
「索引の変更 (Alter index)」ダイ
アログ 69

サマリー・パネル 67
「ターゲット (Target)」パネル
55
「テキスト・プロパティ」パネ
ル 58
「名前」パネル 53

パフォーマンスに関する考慮事項
検索に関する 87
索引作成のための 43

汎用 (GPP) 文書
制限 228
文書フォーマット 23
文書モデルの定義 166

汎用 (GPP) 文書 (続き)
文書モデルの文書タイプ定義
225

ビュー
イベント・ビュー 193
概要 10
作成された ビュー 33
ログ表ビュー 194
db2ext.dbdefaults 33, 187
db2ext.indexconfiguration 33, 192
db2ext.proxyinformation 33, 189
db2ext.textindexes 33, 47, 190
db2ext.textindexformats 33, 192

表スペース 35, 55

ブール演算子
検索構文 148
NOT 83
& (AND) および | (OR) 80

ファジー検索、例 83

フラット ASCII、文書フォーマット
23

付録
インフォメーション・カタログ
187
言語、サポートされている 201
ストップワード 231
大量のメモリーの使用 185
データ・リンク・メッセージ
243
テキスト検索エンジン 231
テキスト検索エンジン理由コード
233
トークン化 231
トラブルシューティング 241
文書モデル 225
マイグレーション 183
メッセージ、シソーラス・ツール
が戻す 249
CCSID 195
Net Search Extender メッセージ
205

文書
構造 165
索引付け 3
サポートされるフォーマット 23
データ・タイプを変換する 36

文書 (続き)
 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
 を交換する 36
 バイナリー・データ・タイプ 36
 DATALINK データ・タイプ 37
文書モデル
 概要 89
 検索構文の SECTION キーワード
 149
 検索構文の属性名 152
 制限 228
 説明 165
 デフォルト 165
 文書タイプ定義 225
 変更 165
文書モデル参照情報 225

[マ行]

マイグレーション 183
メモリー量 185

[ヤ行]

ユーザーの役割
 データベース管理者 25
 テキスト表所有者 25
 DB2 インスタンス所有者 24
ユーザー・シナリオ
 ストアド・プロシージャ検索
 の例 19
 SQL スカラー検索の例 17
 SQL 表値関数の例 20

[ラ行]

列変換関数 9
ログ表
 作成 35
 説明 5
ログ表ビュー 194

ロッキング・サービス
 使用法 27
 表示 29
CONTROL コマンド 100

A

ACTIVATE CACHE コマンド
 構文 113
 使用法 40
AIX でのインストール 14
ALTER INDEX コマンド
 構文 115
 使用法 45
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
 247
 文書コード・ページ 23
 ～のリスト 195
CLEAR EVENTS コマンド
 構文 119
 使用法 46
CONTAINS 関数
 構文 154
 例 78
CONTROL コマンド
 構文 100
 使用法 29
COPYRIGHT コマンド
 構文 144
COUNT キーワード 151
CREATE INDEX コマンド
 構文 121
 使用法 34

D

Datalink Manager
 エラー・メッセージ 243
 データ・リンクの戻りサイズの変
 更 38, 110
DATALINK データ・タイプ 37

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 53
管理 49
 「キャッシュの活動化 (Activate
 the cache)」ダイアログ 75
 「キャッシュ表」パネル 62
 「キャッシュ表の非活動化
 (Deactivate the cached table)」ダ
 イアログ 75
 「更新特性」パネル 60
 「索引イベント」ダイアログ 74
 「索引状況 (Index Status)」ダイ
 アログ 76
 「索引の更新」ダイアログ 73
 「索引のドロップ (Drop Index)」
 ダイアログ 72
 「索引の変更 (Alter index)」ダイ
 アログ 69
サマリー・パネル 67
 「ターゲット (Target)」パネル
 55
データベースの使用可能化 51
データベースの使用不可化 51
テキスト索引の作成 53
テキスト索引の保守 68
 「テキスト・プロパティ」パネ
 ル 58
 「名前」パネル 53
DB2 Net Search Extender の開始
 および停止 50
DB2EXTDL コマンド
 構文 110
 使用法 137
db2ext.dbdefaults ビュー 187
db2ext.indexconfiguration ビュー 192
db2ext.proxyinformation ビュー 189
db2ext.textindexformats ビュー 192
DB2TX、コマンド行プロセッサ
 構文 99, 105, 112
 使用法 31
DEACTIVATE CACHE コマンド
 構文 133
 使用法 40
DISABLE データベース コマンド
 構文 108

DISABLE データベース コマンド
(続き)

 使用法 33, 51

DROP INDEX コマンド

 構文 135

 使用法 46

E

ENABLE データベース コマンド

 構文 106

 使用法 32, 51

EXPAND キーワード 151

F

FUZZY FORM OF キーワード 150

H

HELP コマンド

 構文 142

HTML 文書

 構造化文書 165

 制限 228

 デフォルトの文書モデル 165

 文書フォーマット 23

 文書モデルの定義 169

 文書モデルの文書タイプ定義

 225

N

Net Search Extender

 インスタンス・サービス 27

 開始および停止 27, 50

 キャッシュの活動化 75

 キャッシュの非活動化 75

 更新サービス 30

 索引イベントのクリア 74

 索引イベントをクリアする 46

 索引状況の表示 76

 ストアード・プロシージャ検索
 用キャッシュの作成 38

Net Search Extender (続き)

 データベースの使用可能化 32,
 51

 データベースの使用不可化 33,
 51

 テキスト索引状況を表示する 47

 テキスト索引設定の変更 45, 69

 テキスト索引の更新 43, 73

 テキスト索引の作成 34, 53

 テキスト索引のドロップ 46, 72

 テキスト索引の保守 43, 68

 バックアップおよびリストア 47

 メッセージ 205

 DB2 コントロール・センターの
 使用 49

Net Search Extender インフォメーシ
 ョン・カタログ

 ビューを参照 187

NUMBEROFMATCHES 関数

 構文 155

 例 79

O

OR ブール演算子 80

P

PRECISE FORM OF キーワード

149

R

RESULT LIMIT キーワード 148

S

SCORE 関数

 構文 156

 例 79

search-primary 演算子 149

START コマンド

 構文 102

 使用法 27, 50

STEMMED FORM OF キーワード

150

STOP SEARCH AFTER number

 DOCUMENTS(S)/WORDS(S) キーワ
 ード 148

STOP コマンド

 構文 103

 使用法 27, 50

Sun-Solaris でのインストール 14

T

Table-Valued Function

 ビューに基づくテキスト索引 41

Table-Valued Search Function

 検索 86

 構文 157

TERM OF キーワード 151

U

UNIX でのインストール 14

UPDATE INDEX コマンド

 更新の頻度 44

 構文 139

 使用法 44

 ストアード・プロシージャ用
 40

 RECREATE オプション 44

W

Windows でのインストール 15

X

XML 文書

 構造化文書 165

 制限 228

 デフォルトの文書モデル 165

 文書フォーマット 23

 文書モデルの定義 171

 文書モデルの文書タイプ定義
 225

 XPath 式のセマンティクス 226

IBM と連絡をとる

技術的な問題がある場合は、*トラブルシューティング・ガイド* に示された処置を検討し、実行してから、お客様サポートに連絡してください。このガイドには、お客様サポートがよりよいサービスを提供する際に役立つ、収集可能な情報が示されています。

DB2 Universal Database 製品の情報の入手、または注文については、IBM 担当員に連絡いただくか、認定 IBM ソフトウェア販売業者にご連絡ください。

製品情報

<http://www.ibm.com/software/data/>

DB2 World Wide Web ページには、ニュース、製品説明、研修スケジュールなどの DB2 に関する最新情報が提供されています。ただし、提供されている情報は英語です。

<http://www.ibm.com/software/data/db2/library/>

「DB2 Product and Service Technical Library」では、よくされる質問 (FAQ)、修正内容、資料、および最新の DB2 技術情報などの情報へのアクセスが提供されています。

注: この情報のご提供は英語のみとなりますのでご注意ください。

<http://www.elink.ibm.com/pbl/pbl/>

「International Publications」注文用 Web サイトでは、マニュアルの注文方法についての情報を提供しています。ただし、提供されている情報は英語です。

<http://www.ibm.com/education/certify/>

IBM の「Professional Certification Program」Web サイトでは、DB2 を含むさまざまな IBM 製品の認証テストの情報を提供しています。ただし、提供されている情報は英語です。

<ftp://software.ibm.com>

匿名でログオンしてください。ディレクトリー /ps/products/db2 には、DB2 および多数の他製品に関連したデモ、修正プログラム、情報、およびツールがあります。ただし、提供されている情報は英語です。

<comp.databases.ibm-db2>, <bit.listserv.db2-l>

これらのインターネット・ニュースグループは、ユーザーが DB2 製品に関する自分の経験について話し合うために利用できます。ただし、提供されている情報は英語です。

On Compuserve: GO IBMDB2

このコマンドを入力すると、IBM DB2 Family forum にアクセスできます。すべての DB2 製品が、このフォーラムでサポートされています。ただし、提供されている情報は英語です。

米国以外の国で IBM に連絡する方法については、*IBM Software Support Handbook*の付録 A を参照してください。この資料にアクセスするには、Web ページ：<http://www.ibm.com/support/> にアクセスし、ページの最下部にある「IBM Software Support Handbook」リンク・ボタンを選択します。

注: 国によっては、IBM が承認している販売業者が、IBM サポート・センターの代わりにそれら販売業者のサポート・センターに連絡する場合があります。



部品番号: CT16HJA

Printed in Japan

SH88-8546-00



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: CT16HJA



Spine information:



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 8.1